Practical Electronics

The UK's premier electronics, computing and maker magazine

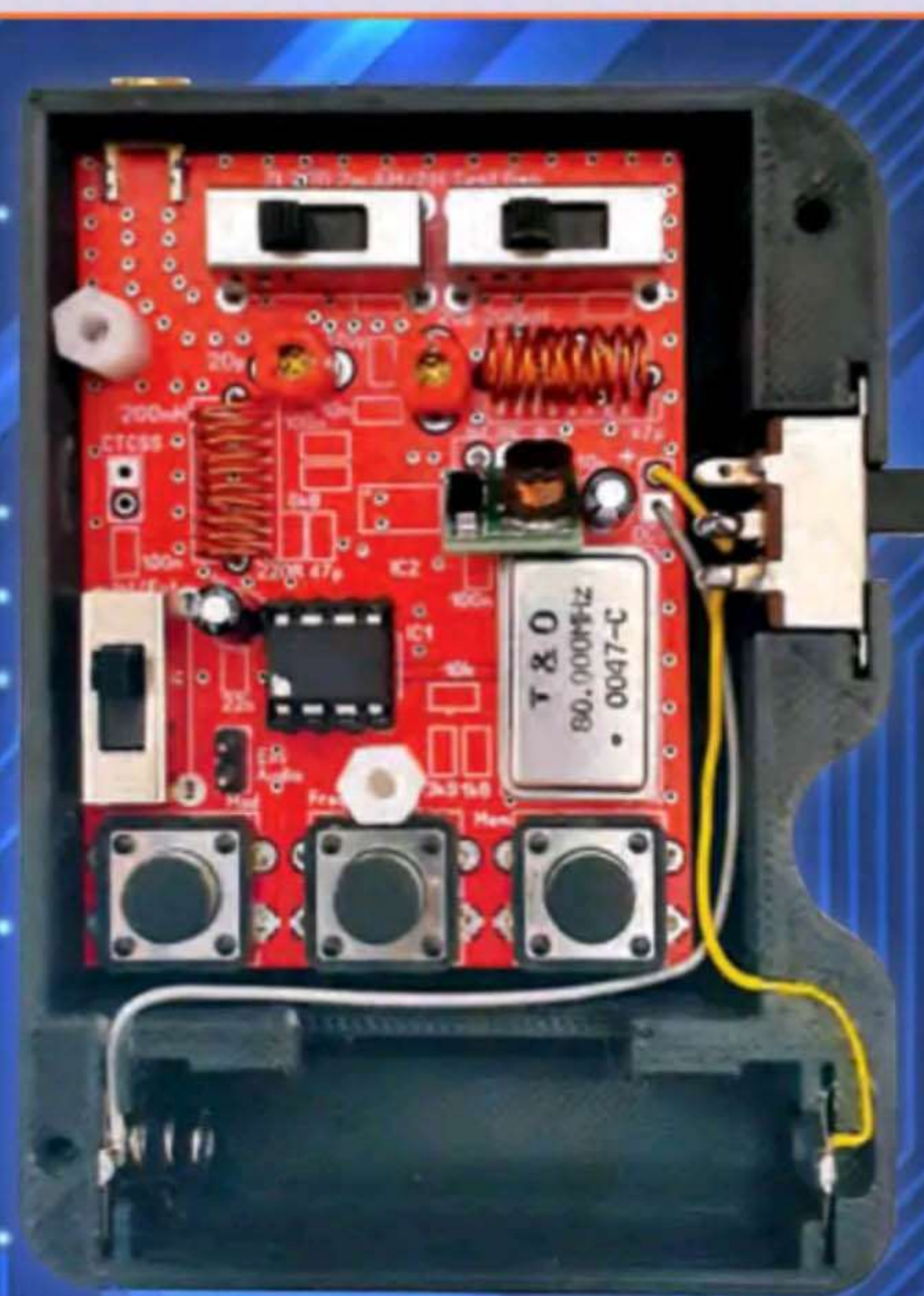
Circuit Surgery
Digital-to-analog converter outputs

Teach-In 2024
Learn electronics with the ESP32, Part 8 – Where Am I? (GPS/GNSS)

Audio Out
More on electronic audio switching

Programming SMD micros

2m band VHF Test Signal Generator



Easy-to-build 500W Class-D Mono Amplifier!

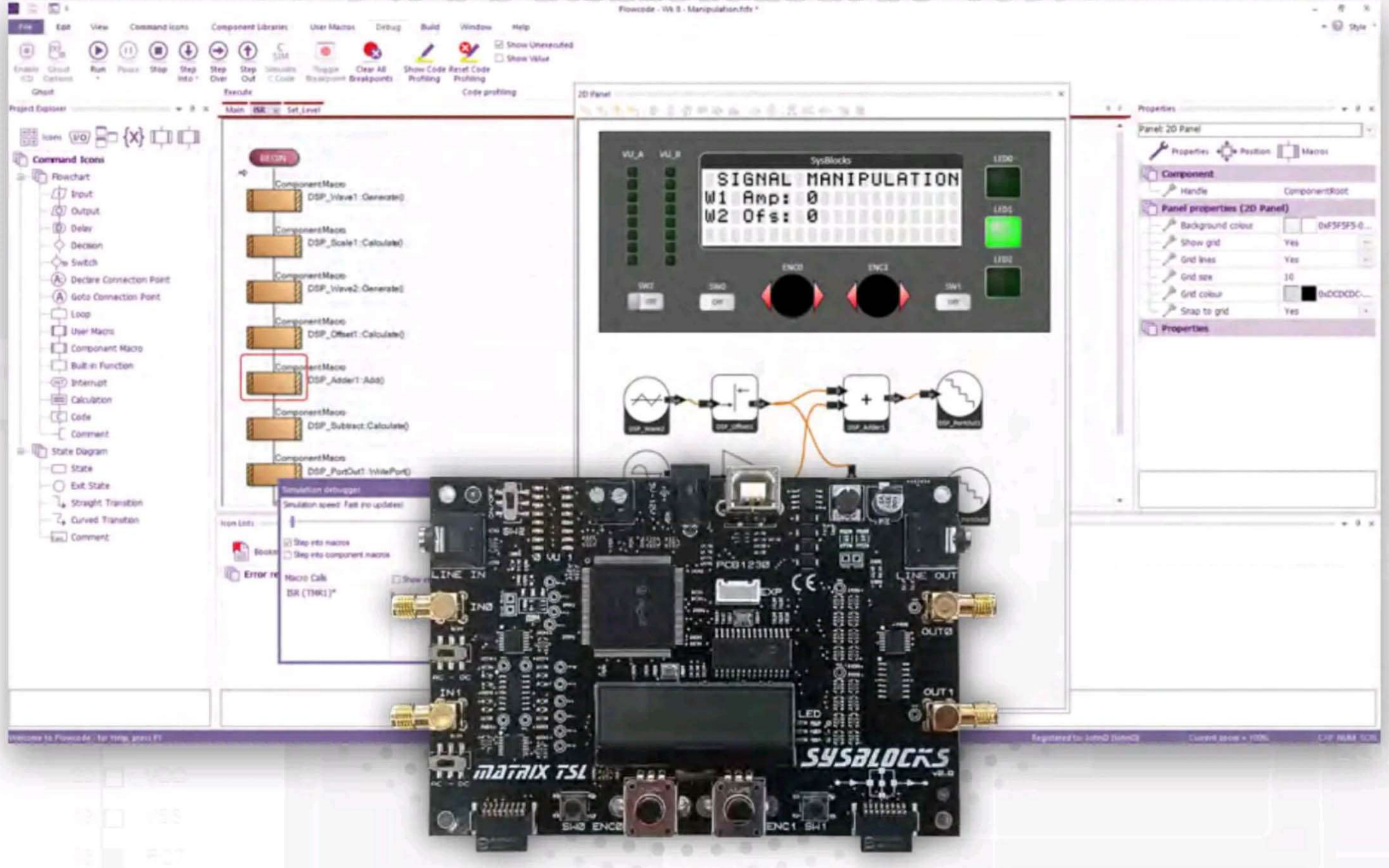
PLUS!

Techno Talk – Sticking the landing (of a model rocket)
Cool Beans – Gray codes and more on BCD handling
Net Work – Low-cost smart watch review

Oct 2024 £5.99

www.electronpublishing.com | @practicalelec | practicalelectronics

Sysblocks


SIGNAL MANIPULATION
W1 Amp: 0
W2 Ofs: 0
MATRIX TSL
SYSBLOCKS
LINE IN
LINE OUT
IN0
IN1
OUT0
OUT1
ENC0
ENC1

CREATE HIGHLY FUNCTIONAL ELECTRONIC SYSTEMS USING PIC32 TECHNOLOGY TO MANIPULATE ANALOGUE SIGNALS USING FLOWCODE 10.1 AND SYSBLOCKS. FANTASTIC FOR DIGITAL AUDIO APPLICATIONS.

Flowcode Pro, PIC32 pack with Sysblocks board, cables and accessories bundle £610 + VAT
Flowcode with PIC32 pack with Sysblocks board, cables and accessories bundle £330 + VAT

10% off for PE readers using code: **PESYS10**

FLOWCODE 10.1 WITH MORE THAN 20 MICROCONTROLLER TARGETS INCLUDING ARDUINO UNO, PIC18877, RPI2B, AND ESP32 IS NOW FREE FOR HOBBYISTS.

☎ +44 (0) 1422 252 380 — ✉ sales@matrixtsl.com

www.flowcode.co.uk

Practical
Electronics

Volume 53. No. 10
October 2024
ISSN 2632 573X

Contents

Projects and Circuits

500W Monoblock Class-D Amplifier by Phil Prosser 8
If you need a serious amount of audio power, are on a budget, and are not after 'high fidelity', this is for you! It uses two prebuilt modules and not much else, mounted in a compact metal chassis, to deliver heaps of audio power all day long.

TQFP Programming Adaptors by Nicholas Vinen 28
Our new Programming Adaptor, described last month, can program chips in DIP and many SMD packages. But what about those in SOT-23-6 or QFP? We show how to program such devices out-of-circuit with several reconfigurable programming jigs.

2m VHF FM Test Signal Generator by Andrew Woodfield 58
This Test Oscillator uses an AD9834 DDS chip in a somewhat unusual way, producing signals in the 2m band (144-148MHz), well above its 80MHz oscillator frequency. It can produce a CW signal and optionally add frequency modulation.

Series, Features and Columns

Techno Talk by Max The Magnificent 3
Sticking the landing – a big win for a high-school rocket scientist

Net Work by Alan Winstanley 4
This month's Net Work checks out what low-cost smart watches can offer, including ideas for devices that help to monitor an individual's health and fitness.

The Fox Report by Barry Fox 7
Hypergram hype

Circuit Surgery by Ian Bell 22
Topics in digital signal processing – digital-to-analog converter (DAC) outputs

Audio Out by Jake Rothman 36
Audio switching Part 5 – real electronic switching, continued

Using Electronic Modules – ADS1115 4-channel, 16-bit ADC by Jim Rowe 42
This small and inexpensive module adds up to four high-speed 16-bit analog-to-digital conversion (ADC) channels to almost any microcontroller via an I^2C serial bus.

Max's Cool Beans by Max The Magnificent 46
Gray codes and more on BCD to 7-segment decoders

Linshang LS172 Colorimeter review by Allan Linton-Smith 54
If you want to find a paint colour that matches your existing paint, this device is perfect. It's relatively inexpensive, very accurate and quite easy to use.

Teach-In 2024 – Learn electronics with the ESP32 by Mike Tooley 68
Part 8 – Where Am I? (GPS/GNSS)

Regulars and Services

Editorial 2
Updates on kits and the magazine

Reader services – Editorial and Advertising Departments 2

Subscribe to Practical Electronics and save money 21

Exclusive Microchip reader offer 27
Win a Microchip dsPIC33A Curiosity Platform Development Board

Practical Electronics back issue downloads 57

Practical Electronics PCB Service 76
PCBs for Practical Electronics projects

Classified ads and Advertiser index 79

Next month! – highlights of our next issue of Practical Electronics 80

Made in Australia & the UK.

Written in Britain, Australia, the USA and Ireland.

Read everywhere.

© Silicon Chip Publications Pty Ltd 2024

Copyright in all drawings, photographs, articles, technical designs, software and intellectual property published in *Practical Electronics* is fully protected, and reproduction or imitation in whole or in part are expressly forbidden.

The November 2024 issue of *Practical Electronics* will be published on Thursday, 3rd of October 2024 (see page 80).

Please note: from 1st of July 2024, subscriptions and renewals paid by PayPal, Visa and Mastercard are now handled via our website:

https://www.electronpublishing.com/

Click the Subscriptions link at the top of the page.

Select Publishing Services will continue to handle renewals via phone, cheque and mail.

Practical Electronics

Volume 53. No. 10
October 2024
ISSN 2632 573X

Editorial

Editorial offices

Electron Publishing (Aust.)	**Tel**	+61 2 9939 3295
Unit 1, 234 Harbord Road	**Email**	pe@pemag.au
Brookvale NSW 2100	**Web**	www.electronpublishing.com
Australia		

Address mail to:

Electron Publishing (Australia)
PO Box 194, Matraville NSW 2036
Australia

Advertising enquiries

+61 2 9939 3295 pe@pemag.au

Editor	Nicholas Vinen
Publisher	Nicholas Vinen
Digital subscriptions	Stewart Kearn **Tel** 01202 880299
Online Editor	Alan Winstanley
Web Systems	Kris Thain
Production	Bao Smith
Technical staff	Tim Blythman, John Clarke

Print subscriptions

Practical Electronics Subscriptions
PO Box 6337
Bournemouth BH1 9EH **Tel** 01202 087631
United Kingdom **Email** pesubs@selectps.com

Technical enquiries

We regret that technical enquiries cannot be answered over the telephone. We are unable to offer any advice on the use, purchase, repair or modification of commercial equipment or the incorporation or modification of designs published in the magazine.

Questions about articles or projects should be sent to the editor by email: pe@pemag.au

Projects and circuits

All reasonable precautions are taken to ensure that the advice and data given to readers is reliable. We cannot, however, guarantee it and we cannot accept legal responsibility for it.

Some projects and circuits published in *Practical Electronics* employ voltages that can be lethal. Do not build, test, modify or fix any mains-powered equipment unless you fully understand the safety aspects involved and you use an RCD (GFCI) adaptor.

Component supplies

Silicon Chip Publications may offer kits or other parts for making our projects, but not in all cases. When kits are not available, readers will need to find and source parts themselves. We advise readers to check that all parts are still available before commencing any project in a back-dated issue.

Advertisements

Although the proprietors and staff of *Practical Electronics* take reasonable precautions to protect the interests of readers by ensuring as far as practicable that advertisements are bona fide, the magazine and its publishers cannot give any undertakings in respect of statements or claims made by advertisers, whether these advertisements are printed as part of the magazine, or in inserts. The Publishers regret that under no circumstances will the magazine accept liability for non-receipt of goods ordered, or for late delivery, or for faults in manufacture.

Transmitters/bugs/telephone equipment

We advise readers that certain items of radio transmitting and telephone equipment which may be advertised in our pages cannot be legally used in the UK. Readers should check the law before buying any transmitting or telephone equipment, as a fine, confiscation of equipment and/or imprisonment can result from illegal use or ownership. The laws vary from country to country; readers should check local laws.

Updates on kits and the magazine

Now that Silicon Chip Publications is producing this fine magazine, I decided that we should overcome the difficulties we have been having since 2021 in selling items like kits to customers in the UK. That was when the UK government decided to change VAT laws in a way that made it very difficult to sell to customers in the UK from outside.

We made the necessary changes to our website and procedures to sell to UK customers in early July. Unfortunately, we are still waiting for the UK government to give us the go-ahead. It seems like a silly situation to be in, but we must follow the applicable laws. Unfortunately, that means waiting for the gears of officialdom to turn, and they turn very slowly...

I apologise if any readers saw offers of kits in the September issue and have not been able to purchase them yet. Rest assured that you will be able to do so as soon as we are given the go-ahead. I hope that they will be available to purchase before this issue even reaches you; we shall see.

Besides that 'red tape' delay, it seems that the changeover has gone pretty smoothly. Readers appear to have figured out the new online renewal process via our website. We have also spent quite a lot of time and effort trying to clean up the Practical Electronics online shop, which was missing quite a few items, had incorrect stock levels for various products etc. It's still a work in progress but we feel that it's in much better condition now than when we took over.

I must thank Matt Pulzer of Electron Publishing for being very helpful in guiding me during this transition period. We have plenty of experience in publishing magazines, but each magazine has its own quirks and procedures and they would have been very difficult to navigate without the assistance we have received.

You may notice that Net Work is considerably shorter in this issue that it has been in the past. Alan Winstanley has decided, after doing this for a long time, that he wants to cut back on the time he spends writing for *PE*. I'm happy he has decided to continue writing for us, even if it's in a reduced capacity. Something is better than nothing!

We have managed to fit an extra couple of articles in this issue on topics that I think *PE* readers will enjoy. Please feel free to send feedback (positive, negative or other) to the email address at the top of the sidebar.

Nicholas Vinen, Electron Publishing (Australia)*
Publisher & Editor,
Practical Electronics Magazine

* a division of Silicon Chip Publications Pty Ltd.

Sticking the landing

Techno Talk

Max the Magnificent

Doctor Who once brought Vincent van Gogh to our time to see an exhibition of his work and to hear himself described as "the greatest painter of them all". If I ever get my own time machine working, I'd love to do something similar for many of the great inventors.

Most of the time, we focus our attentions on the wonders of the latest and greatest in existing and emerging technologies. So it's easy to forget that there's still a lot of old technology lurking about.

In industry, for example, we beguile ourselves with the idea of smart automation and smart machines. We visualise things like pumps, generators and motors being equipped with a bevy of advanced sensors feeding some form of artificial intelligence/machine learning (AI/ML) that can perform tasks like detecting anomalous behavior and predicting problems.

However, there remain myriad factories around the world that are jam-packed with machines that are as dumb as a pile of rocks (but still quite effective).

It isn't just industry either. The US Air Force's intercontinental ballistic missile command, control and communications network stored critical data on 8-inch floppy disks until 2019.

You think that's bad? I just heard that as part of the German Navy's current modernization efforts, they've decided their Brandenburg-class F123 frigates are no longer going to rely on 8-inch floppies as part of their onboard data acquisition (DAQ) systems. However, rather than overhauling the DAQs in their entirety, the solution they've come up with is to create subsystems that emulate 8-inch floppies. I don't know whether to laugh or cry!

How big? How heavy?

I once had an interesting chat with Horst Zuse, the son of Konrad Zuse. As a young man, Konrad single-handedly built the world's first fully mechanical binary floating-point computer, which he modestly called the Zuse 1 (Z1).

By 1938, this bodacious beauty was up and running in his parents' family room in Berlin. Konrad went on to create the world's first working programmable, fully automatic, Turing-complete computer.

Known as the Z3, this beast featured mechanical memory and relay-based logic. The Z3 was fully operational in 1941, which means the Germans were well ahead of the Americans and British on the computing front.

Konrad's next effort was the Z4. This was only partially completed when WWII really started to go pear-shaped for the Germans, resulting in Konrad and his Z4 being evacuated from Berlin to a bunker in the Hartz mountains in northern Germany.

Can you guess who Konrad ran into in this bunker? It was none other than aerospace engineer Wernher von Braun. Wernher was then working on his V2, the world's first long-range guided ballistic missile. The V2 rocket became the first artificial object to travel into space on 20 June 1944. After the war, Wernher moved to America where his work was key to the success of the Mercury, Gemini, and Apollo programs.

During our chat, Horst and I contemplated the conversations that may have taken place between Wernher and his father. Wernher would have loved a small, light, computerized control and guidance systems that would fit in his rockets. But Konrad's computers were the size of rooms and weighed tons.

Neither Konrad nor Wernher could have conceived of concepts like semiconductor transistors and integrated circuits. If only they could see us now!

Rocket boy

Did you ever see the 1999 American biographical drama film October Sky? This tells the true story of a coal miner's son called Homer H. Hickam Jr.

Inspired by the launch of Sputnik 1 in 1957 and against his father's wishes, Homer and his friends started to design and build model rockets. They won their school science fair and participated in the National Science Fair in Indianapolis.

Homer eventually became an engineer working with Wernher von Braun at NASA's Marshall Space Flight Center in Huntsville, Alabama, just a few miles down the road from the office where I'm writing this column.

The Eagle has landed

I remember seeing the first landing on the moon. This was in 1969, when I was 12. I still choke up when I hear a recording of Neil Armstrong saying, "Houston, Tranquility Base here. The Eagle has landed". I also remember feeling sad when I learned that the first and second stages of the Saturn V super heavy-lift launch vehicles used in the Apollo missions fell into the ocean after separation.

As a young lad, I read a lot of science fiction books, like *Rocket Ship Galileo*, *Space Family Stone* and *Space Cadet* by Robert Heinlein.

Maybe that explains why I was so excited and enthused in 2015 when Blue Origin and SpaceX both demonstrated vertical takeoff, vertical landing (VTVL) and recovery of their launch vehicles after return to the launch site (RTLS) operations with their New Shepard and Falcon 9 rockets, respectively.

Did you ever see the *Young Sheldon* episode where a representative from NASA gives a presentation to Sheldon's class at school? Sheldon asks why NASA doesn't employ VTVL. The NASA guy says the mathematics is too complicated, so Sheldon opens a new notebook and starts writing.

Once he's finished, Sheldon persuades his dad to drive him to NASA's Johnson Space Center in Houston, where he presents his notebook to the guy who said it wasn't possible, after which it's "job done" as far as Sheldon is concerned.

At the end of the episode, we jump forward in time to a news report showing a SpaceX Falcon 9 performing a vertical landing. Then the camera cuts back to Elon Musk in his office with Sheldon's notebook in his hands. When his secretary announces a visitor, Elon quickly slides the notebook into his desk drawer.

Sticking the landing

I rarely remember what inspires my columns, but in this case, it was an item on the Hackaday website that set my juices flowing (**https://pemag.au/link/abzc**).

There I discovered that, while attending Montgomery High School in New Jersey, a student called Aryan Kapoor recently managed to do something that would have made Homer Hickam and Wernher von Braun squeal in excitement.

After three years of research, design and testing, Aryan managed to achieve vertical takeoff and vertical landing with a model rocket. This was all made possible by modern sensors and actuators. It makes me wonder what school kids will be doing 100 years from now! PE

Net Work

Alan Winstanley

This month's Net Work checks out what low-cost smart watches can offer, including ideas for devices that help to monitor an individual's health and fitness.

The idea of smart watches is hardly new – horologists have long sought to add features or so-called 'complications' to simple clock and watch movements, to display the date, elapsed time or maybe the phase of the moon, for example.

The advent in the early 1970s of the digital wristwatch brought with it solid-state reliability, continuous and precise timekeeping and electronic 'complications' including piezo alarms, calendar readouts and stopwatch timekeeping.

Swiss watchmaker Hamilton is credited with producing the first quartz watch with a seven-segment LED display, the 1972 Hamilton Pulsar, as seen in contemporary James Bond movies. A fascinating history of them is recounted at https://pemag.au/link/abzo

In the same year, the Swiss company SGT (Société des Gardes-Temps) created a digital watch movement that used a liquid crystal display (as told in Electrifying Time by M.E. Theaker BSc, *Practical Electronics*, May 1978). A 1972 Waltham-branded watch containing such a movement is displayed at the Science Museum website at https://pemag.au/link/abzp

10:09
HAMILTON

Timeless (?) fashion: a modern take on the original Hamilton Pulsar LED watch from half a century ago is still on sale.

The clear, always-on battery-powered digital display was a great novelty at the time. A 1976 iteration of this concept by SGT is shown on the Europastar website at https://pemag.au/link/abzq

Look ma, no hands!

Hamilton still produces an updated and pricey 'American Classic PSR' that they list at https://pemag.au/link/abzr

A favourite wristwatch of mine is the Bulova Computron, a more affordable remake of their classic LED digital watch from the 1970s. For UK readers, at the time of writing, online jewellers H. Samuel sells it – but hang back for discount offers. More details are at https://pemag.au/link/abzs

In July 1978 Texas Instruments launched its seminal TI 'Starburst' LCD watch, a new kind of timepiece with 'electronic hands' formed from liquid crystal polygons instead of digits. I found photos of a Starburst hosted by the Southern Methodist University (SMU) in Texas at https://pemag.au/link/abzt

The December 1979 issue of *Practical Electronics* gave readers an opportunity to own one of these new breed of technological masterpieces. Seiko's new 'digital analogue' watch combined a digital LCD readout with a separate quartz analogue watch at an offer price of £79.95, or about £385 in today's money. Then came full-function calculator watches that used a 'button operating tool' to press buttons on a tiny keypad. The quartz-powered watch was here to stay!

The Bulova Computron is a more affordable modern-day retro LED watch. Different case and LED colours are available.

Fast forward to the 21st century, and digital electronics, display and sensor technologies have advanced to such an extent that powerful 'wearable' devices such as smart watches have become universally popular. Built-in health monitors or 'fitness trackers' are commonplace and some high-end ones incorporate GPS as well.

Using Bluetooth, smart watches also put common phone functions within easy reach on your wrist. Even my low-tech plumber glances at his smart watch to see who's calling on his phone; he still ignores them anyway!

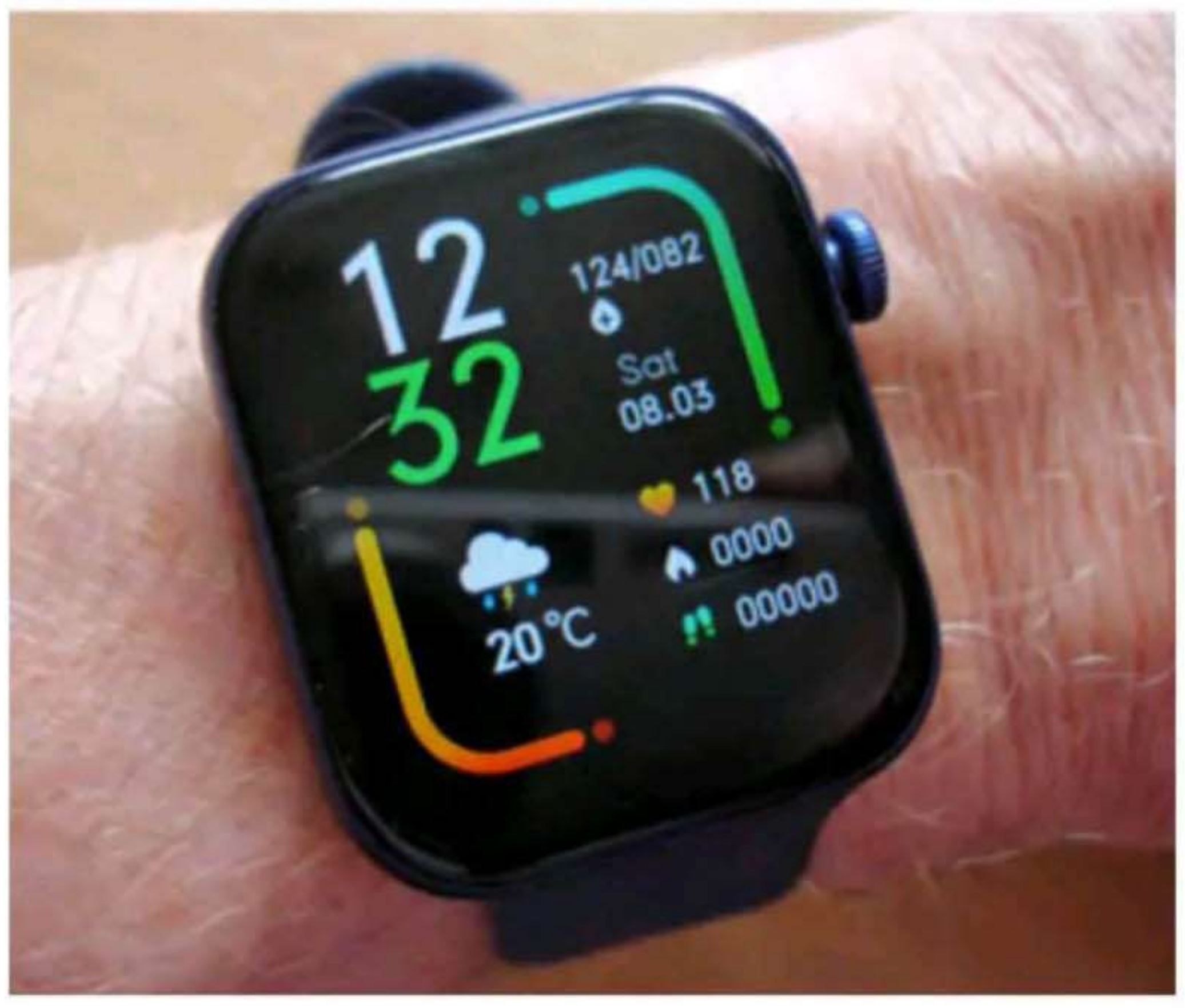
Time to work smartly

I decided to investigate what a low-cost smart watch can offer as readers may not have considered owning one before. Many smart watches retail at £30, with some even under £20.

A cheap and cheerful £30 model duly dropped on the doormat and, as expected, it was typical Chinese-made fare, unbranded except for 'Acclafit', plus a tiny email address and very little else.

It included a decent manual that pointed to the customary smartphone app ('H Band'), which was soon installed. I declined the offer of uploading my data to a cloud server, but, unlike with some products, the privacy implications were spelled out clearly in advance. I also received a helpful response from the support email address printed on the box, so it wasn't all bad news.

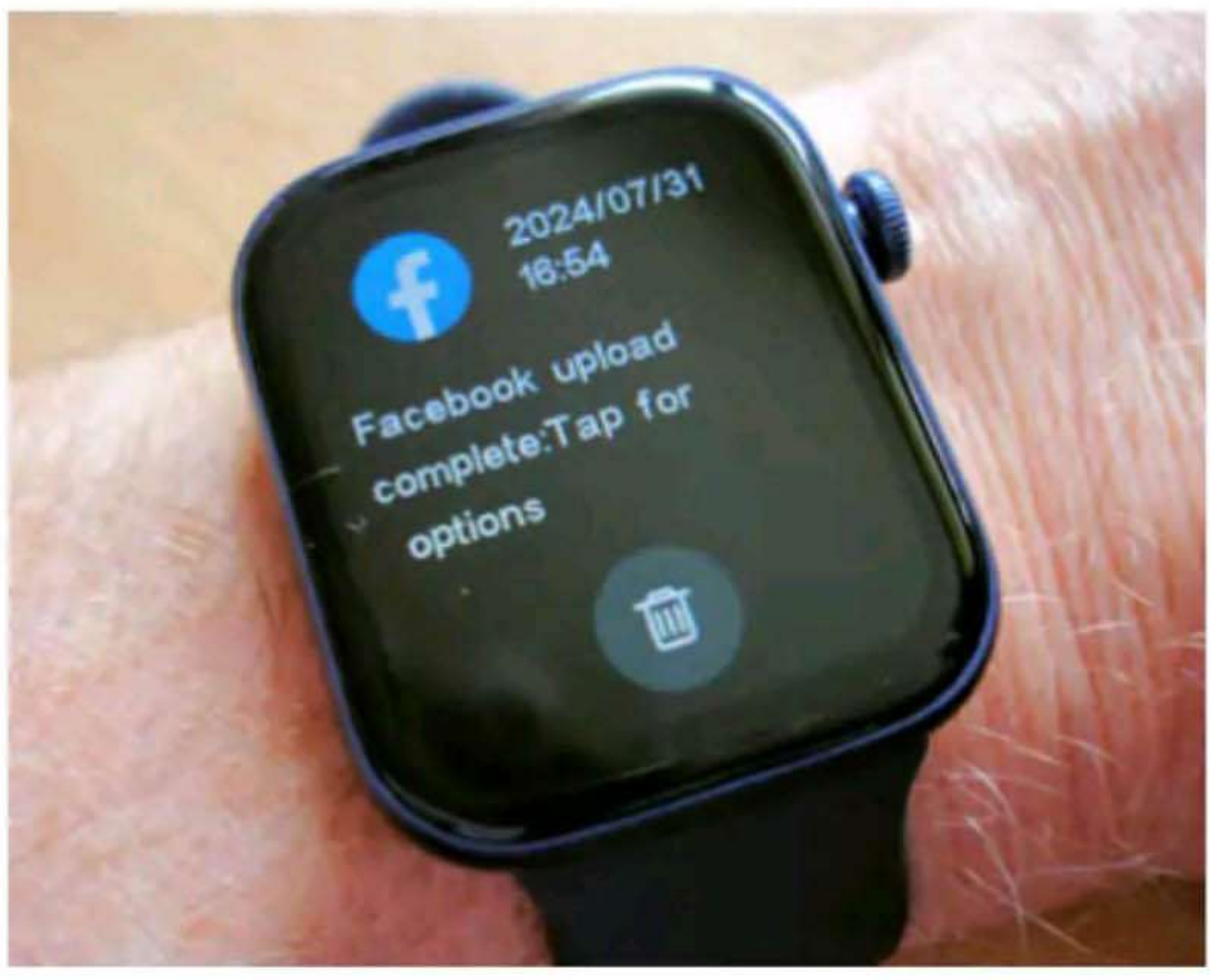
I have to admit that this budget device



This is the low-cost smart watch I tested, packed with fitness-tracking features. The OLED display is clear and bright.

SMART WATCH

The rear LED array uses the PPG principle to monitor the wearer's bloodflow for pulse rate, blood pressure readings and more.



2024/07/31
18:54

Facebook upload complete:Tap for options

Facebook and Whatsapp messages can appear on the smart watch, and a phone dialler is also available.

BRAUN
96
SpO2 %
66
PR bpm

A Braun Pulse Oximeter uses an LED to display SpO_2 oxygen saturation and pulse rate on an OLED. All good here!

far exceeded my expectations, especially given the low price. The OLED display is crystal clear, colourful and viewable outdoors even in bright light. A tilt of the wrist lights the OLED screen for a short time (you can disable this automatically, eg, at night) and the waterproof strap is comfortable to wear. The smooth-edged, metallic-finish watch case is easy on the wrist as well as the eye, and the push button 'crown wheel' works well to fire up the display. Early impressions of the watch were very promising, with nothing to dislike about it at all.

You can select different watch faces (more are available via the app) or upload a favourite photo from your phone instead, something I enjoyed. Even the weather and temperature can be shown as data is fetched from the smartphone. A 'torch' mode sees the display glow white, which might sometimes be better than having nothing at all.

Well that's a long word!

The fitness fun continues, aided by the watch's built-in monitoring features. On the underside, several small LEDs in snug contact with the skin are used to sense the wearer's blood flow. *Photoplethysmography*, or PPG, involves shining green and red/infrared light into the skin and using the reflections to measure changes in volumes of your blood circulation as the heart pumps blood. One's heart rate and blood oxygen levels can then be estimated.

SMART WATCH

A USB charging lead simply clicks onto the case rear using magnets.

Green light is absorbed by red blood, though, controversially, this technique may be inaccurate for those with darker skin.

The surprising specifications rack up, as the OLED is also a touch screen that lets you swipe in any direction to access submenus while a vibration motor generates alerts.

The watch is recharged via a USB lead that clicks magnetically on the case rear. This works well enough, though you could conceivably short out the power supply accidentally with a paper clip, metal swarf or staple if you're not careful. Commendably, despite the small size, the battery lasts several days.

This particular watch will also have a go at counting steps as you target 10,000 a day, and it will check your sleep pattern, measure the wearer's blood glucose (in mmol/L) and take blood pressure too. The smartphone app amasses three days' worth of data in colourful graphs.

You can use Microsoft's Phone Link app to mirror your smartphone screen and view them on your Windows PC monitor as well.

Some features rely on the watch being within Bluetooth range of your phone. The OLED display will then update with new Facebook and Whatsapp messages, which proves handy if I'm busy working. The watch vibrates when the phone rings, and it can store favourite contacts, with a keypad being available for dialling out.

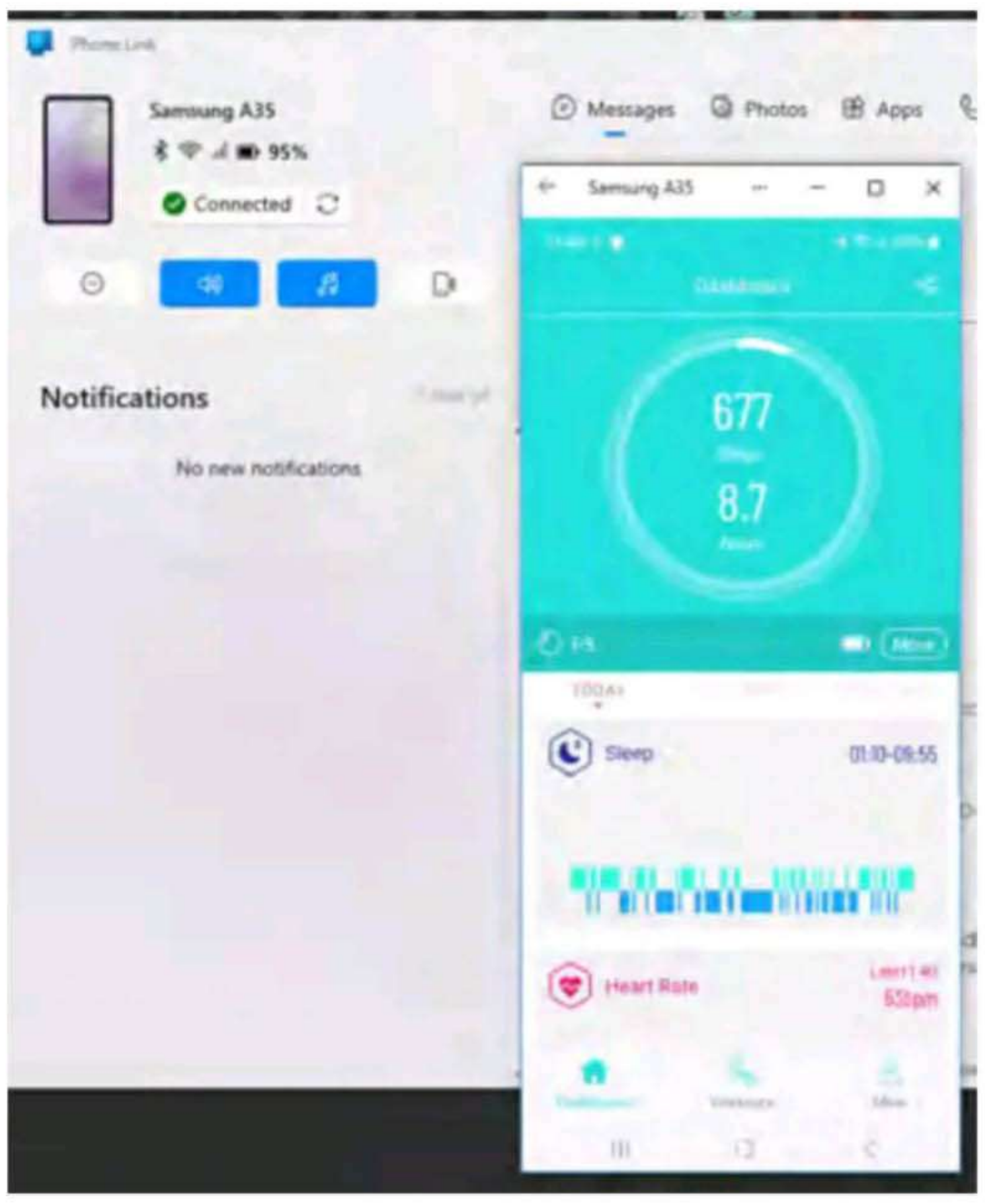
Overall, the makers have packed in many features that can complement the way you work, or help if you want to check some fitness factors without constantly grappling with a smartphone or a medical device. It's well-made and feature-packed, but the big question is: how accurate is a cheap smart watch like this?

One can't expect laboratory standards, but it's probably on a par with many budget priced cuff-type sphygmomanometers (blood pressure measurement devices), some of which cost two or three times more but can be extremely difficult to use.

In my case, the smart watch displayed a BP reading of 124/82 mmHg (not bad, despite everything!). Out of interest, I compared it against a Braun iCheck 7, an inflatable cuff-type blood pressure monitor that records data on an app stored on my tablet. Two readings taken using the iCheck 7 were 116/77 and 122/81, both pretty close to the watch's readings. When values vary wildly, medical experts recommend settling down and taking an average of multiple readings, allowing a minute or two in between.

There's more choice than ever of electronic devices that keep track of our health and fitness, including a Braun Pulse Oximeter that clips onto a fingertip and displays both oxygen saturation (SpO_2) and heart rate on a colour OLED screen.

You can track body weight with, for example, the Etekcity-brand Smart Bathroom Weighscales (£20+) which record BMI, visceral fat and many other factors. I use one with an app on a tablet to track my weight management.



The Microsoft Phone Link app lets you operate and view your Android or iOS smartphone from a Windows PC.

By capturing results over time, these kinds of devices can assist in generally gauging trends in one's health and fitness. A wide variety of them is available on the usual web sites. A low-cost smart watch could be a surprisingly useful accessory that's worth experimenting with, especially at these meagre prices.

Cazoo goes under

There was a time when buying a motor car meant trawling around showrooms, taking test drives, haggling over trade-ins and drooling over glossy brochures back at home. My dad did it all the time!

More recently, Britain's car sales sector hit on the idea of selling cars directly to customers using virtual showrooms instead, showcasing shiny used cars online almost as if they were new vehicles. Simply click to buy online and one will be delivered on a trailer to your door, along with a money-back guarantee!

Personally, I never believed anybody would want to make such a huge investment without seeing a car in the flesh or taking it for a spin. Reality has finally prevailed, and online trader Cazoo – once valued at $5 billion and whose TV car sales ads (eg, https://pemag.au/link/abzu) reached saturation level at one point – has gone out of business and filed for administration, after they finally ran out of road and hit a brick wall.

Chinese-owned MG Cars continues its upwards sales trajectory buoyed by its range of keenly-priced electric cars, as highlighted extensively in earlier articles. UK consumer group *Which?* recently reported serious issues with MG4 cars, when they claimed lane-keeping and lane departure systems twice pulled their test MG4 onto the wrong side of the road.

Which? reported the MG4 also steered towards an oncoming vehicle on a country lane. According to *Which?*, some MG4 owners have even handed their cars back, but MG said that a software update was being developed and reckoned the problem was 'not a safety issue', before refusing to comment any further.

You can read the consumer body's report and decide for yourself at https://pemag.au/link/abzv

Britain's car sales network was recently polled by trade magazine Car Dealer for their views about working with their particular vehicle franchise. In the Car Dealer Power 2024 survey, coming 25th at the bottom of the list was... MG Cars.

It seems the honeymoon might be over as, according to the survey, traders criticised MG's aftersales, forward planning, accessibility to senior management and (whisper) sales bonus structure. Top of the list was Suzuki. The full survey is at https://pemag.au/link/abzw

Space update

At the time of writing, Boeing's Starliner capsule remains long overdue for returning to Earth, having docked for a brief visit at the International Space Station in early June. Further tests, simulations and inspections on flight systems are being carried out before the capsule can undock and return its two occupants safely, "when a date is chosen", Boeing says.

In July, four NASA scientists successfully completed a year-long simulation of what life on Mars might really be like for humans living in the confines of a Martian-like habitat.

The CHAPEA (Crew Health and Performance Exploration Analog) mission is the first of three such simulations, and you can see how the first mission was constructed and ran – 3D-printed buildings and all - at https://pemag.au/link/abzx

Lastly, the November issue will mark 60 years of *Practical Electronics* magazine, so in the next issue I'll look back at the story of PE and how Internet connectivity has shaped our world since those early days of 'dial up'. There's plenty to look forward to! PE

Colette Brownrigg

Long-time readers will recall the classic style of Everyday Practical Electronics throughout the 1990s, when every article carried a sketch alongside the heading. They were drawn by Colette Brownrigg, part of our team and a gifted designer and freelance artist whom I knew for 30 years.

In July, I received the sad news that Colette had passed away very suddenly, just three weeks after her husband, whom she had been caring for, had also passed away after a long illness.

A dear friend and confidante, Colette will be greatly missed by her many friends and former co-workers, and we extend our sincerest condolences to her daughter and family.

HAMMOND®

1556 FR ABS IP54 enclosures

Learn more:
hammondmfg.com/1556

uksales@hammondmfg.com • 01256 812812

The Fox Report

Barry Fox's technology column

Hypergram hype

Victoria rail station in London often stages pop-up promotions. Usually they are product sample giveaways – a free packet of biscuits or canned drink is thrust at everyone rushing for a train. But sometimes a new gadget, for cleaning or cooking or somesuch, is on show.

Recently, I stopped dead in my tracks and nearly missed my train when I saw what looked like a hologram floating in the concourse space. It was brighter and more colourful than any hologram I've ever previously seen.

The company behind it, Disturbed Media, is in the attention-grabbing business. Their web site is very long on hyperbole and short on hard fact: **https://www.disturbedmediagroup.com/hypertouch/**

Buried in the promo-gloss you'll find reference to a system called Hypergram, with clear reference to 'hologram' technology but virtually zero technical fact:

The future of holographic technology

Our hypergram experiences utilise the latest holographic technology, revolutionizing 3D experiences by bringing holographic experiences to life. Without the need for a screen, these 3D objects can literally float in mid-air and be perceived as truly 3-dimensional.

Whether it's a product, a person, or a complex 3D story, our hypergram experiences makes it possible. With no boundaries and unlimited potential for 3D projects, we will transport you into new worlds of 3D creative exploration.

Hypergram has its own website (**https://www.ingagemedia.co.uk/hypergram**), with similar opaque gloss:

Hypergram brings you the future of OOH consumer entertainment and engagement.

Specifically engineered for the human eye, Hypergram is the latest innovation in HD Holographic technology.

Viewable without the need for any apps or glasses, Hypergram makes objects float in the air, coming out of the screen and towards the viewer.

Obviously, like stage magicians, the companies promoting the technology want to create an air of mystery. So it's understandable that their websites give away no technical secrets, or names of inventors to make patent searching eaiser. Of course it's usually possible, given the time and need, to search out any patent for anything (other than classified military technology, which can be sealed from public view).

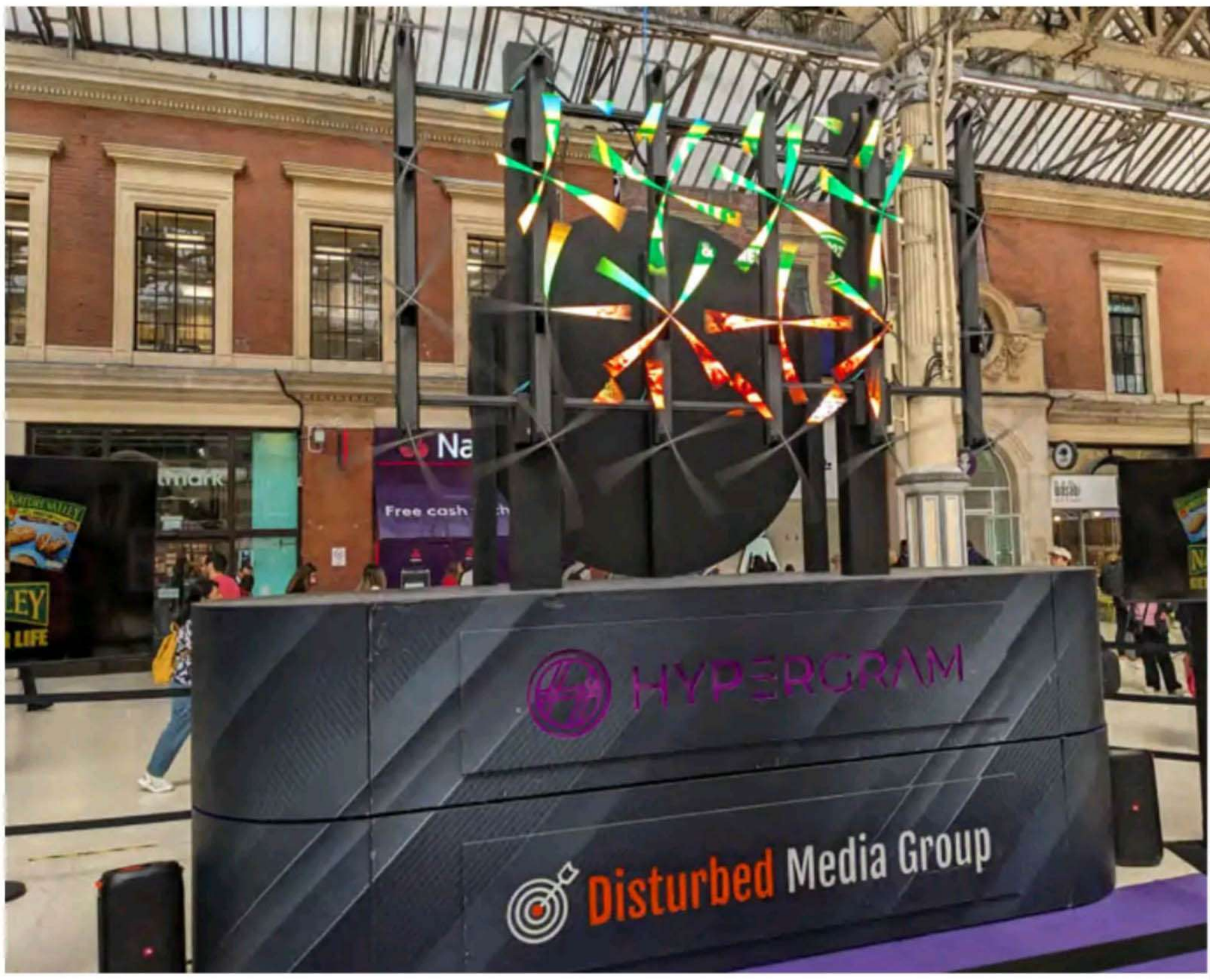
In this case, there is no need to go that route because one of the demonstrators at Victoria was perhaps naively forthcoming. Also, fast shutter photos reveal much of how the system works.

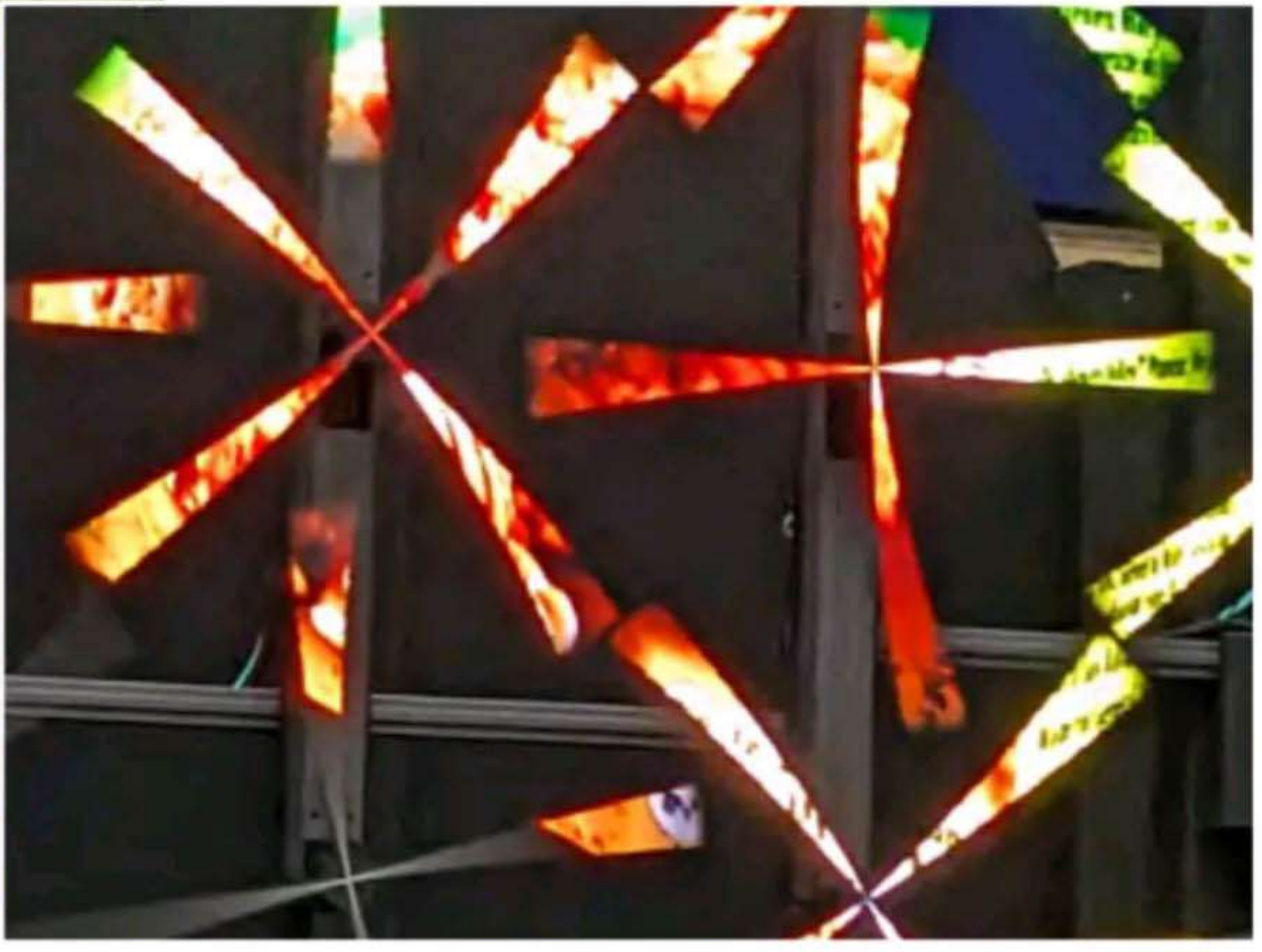
The 'hologram' is created by an array of fans, with long, straight, dark strip blades that carry very bright coloured LEDs. The blades spin fast as the LEDs are triggered by appropriately switched signals from a laptop operated from alongside the rig. The blades rotate fast enough to become invisible to the human eye, while the LEDs paint a bright video picture, apparently hanging in space.

The idea is ingenious and works well; far better than previous systems that relied on projecting conventional video images on fog, water mist or semi-transparent screens. I guess we will soon see it used at music events and stage shows. Hobbyists may want to experiment with similar ideas.

The system may neatly combine the invisible fan concept with the increasingly-often seen flexible LED screen 'wallpaper' that lets a bare theatre stage set become a city, landscape or spacescape at the flick of a switch. The rotating fans could perhaps be coated with video wallpaper.

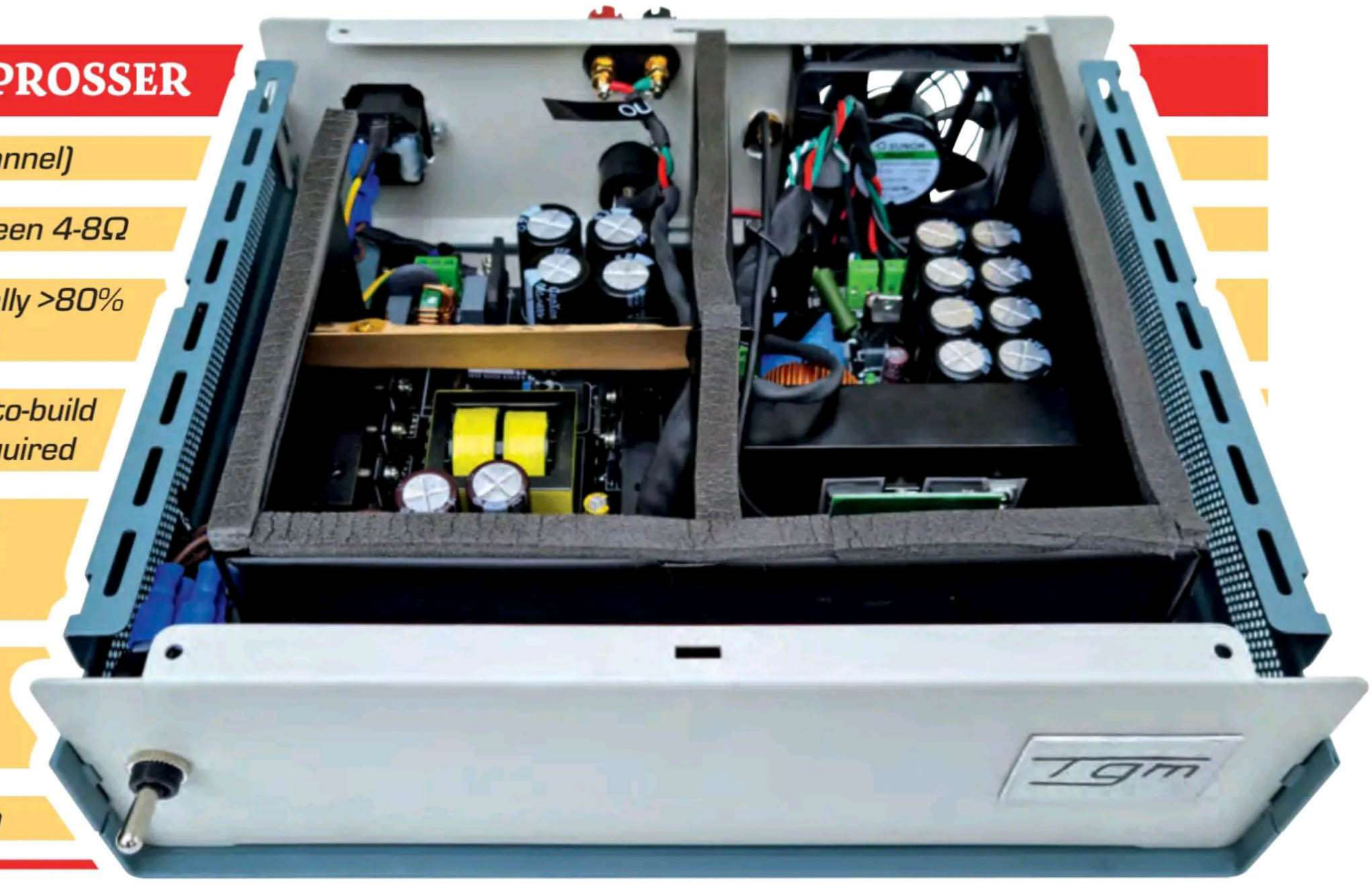
The Menier Chocolate Factory theatre in London was one of the first UK venues, a few years ago, to use this kind of active wallpaper.





DESIGN BY PHIL PROSSER

- *485W into 4Ω (single channel)*
- *Operates with loads between 4-8Ω*
- *Very high efficiency (typically >80% at moderate power levels)*
- *Very low in cost and easy-to-build with minimal soldering required*
- *Typically, 0.02% distortion over most power levels at 1kHz*
- *Frequency response from <5Hz to 20kHz, +0,-1.5dB*
- *Built-in speaker protection*



500 WATT MONOBLOCK CLASS-D AMPLIFIER

If you need a serious amount of audio power, are on a budget, and are not after 'high fidelity', this is for you! It uses two prebuilt modules and not much else, mounted in a compact metal chassis, to deliver heaps of audio power all day long.

Building a 500W+ amplifier is a serious undertaking. To make sense, a high-power Class-D amplifier would need a switch-mode power supply. After all, why bother with a Class-D amplifier if you need a 1kVA transformer and bank of capacitors, making the thing half the weight of a VW Beetle?

So we came up with the idea of using some of the relatively cheap modules available on sites like eBay and AliExpress.

There were three questions on our minds: were they safe, would they even work, and would the performance be acceptable? So we started surfing online shops and came across two promising modules (see the adjacent panel).

Deciding on the modules

All the modules we purchased for evaluation have reasonably good availability and have been on sale for many months. Some things that drove us to choose them were:

DANGER – LIVE COMPONENTS

Do not consider touching the heatsinks or anything on the PCBs when the amplifier is powered or for several minutes afterwards. Assume that contact will be lethal! Never, ever touch the PCBs if the amplifier is even plugged in. If you want to measure the heatsink temperature, use a non-contact IR thermometer.

Table 1 – Measured performance into a 4Ω resistive load

Voltage (RMS)	Freq	Load	Power	THD+N	Notes
8V	1kHz	4Ω	16W	0.026%	Warm up test, heatsinks 36°C
20V	1kHz	4Ω	100W	0.017%	Heatsinks 40°C after a few minutes
30V	1kHz	4Ω	225W	0.019%	Heatsinks 49°C after a few minutes
40V	1kHz	4Ω	400W	0.03%	
44V	400Hz	4Ω	484W		Output started clipping
44V	1kHz	4Ω	484W		Output started clipping

- The power supply modules have decent mains-to-secondary isolation.
- They have decent heatsinking and quality capacitors.
- They are common/available parts sold in a range of voltages, ie, a moderately mature and supported design.
- The prices are neither too cheap to be true nor overly expensive.

So we placed orders for one of each to test out (plus the modules listed in the panel overleaf that we didn't end up using). The cost of each module was in the region of $100.

$200 for a power supply and amplifier module is bonkers for this sort of power level. If you have built a 500W amplifier using discrete parts and a linear power supply, you will know that this would barely pay for the transformer, let alone the rest.

So is this still too cheap to be true? Our greatest concern with purchasing this sort of equipment online is safety and electrical standards. In choosing these modules, we spent a lot of time downloading photos and trying to see how they were laid out, if there were slots milled between feedback opto-isolators and suchlike.

Once we had received them, we inspected them to see if they matched the pictures – they did. We then tested them to the best of our ability using our old-school megger (500V) and found no measurable leakage from primary to secondary on both power supplies during a 60-second test.

We are not promoting these power supplies as being compliant with any standard, mind you! But there is visible isolation built into the design and measurable isolation on test, which was enough for us to work with them.

The power supplies purchased both claim to be capable of "1000W", although the smaller of the two owns up to being more like a 500W continuous unit. We think it reasonable to rate both power supplies under 1kW continuous, given the parts used, especially the smaller one.

Of the two sets of modules, we chose to proceed with the larger, black modules. We have provided some information on the 'also-ran' modules for interest but recommend that you stick with the two shown opposite.

Performance

Table 1 shows some spot measurements of distortion at various power levels. These agree with the claims

#1 Large Class-D Amplifier

IRS2092S 1000W Class-D Mono Amplifier (see Photo 1):
pemag.au/link/abic
pemag.au/link/abid
pemag.au/link/abie

Claims

- Speaker protection operating from an independent power supply
- Supply voltage: ±65V to ±80V
- Maximum output power: 1000W
- Efficiency: ≥90%
- Signal-to-noise ratio (SNR): 90dB
- Dimensions: 157 × 101 × 44mm
- Net weight: 0.45kg

Photo 1: this 500W+ Class-D amplifier module was under $100 and includes a speaker protection relay. The control circuitry is mounted on a vertical sub-PCB.

This has four onboard 1000μF 100V supply bypass capacitors per rail labelled Nichicon HE(M), 18mm in diameter and 42mm high. The Nichicon data sheet we found did not list a 1000μF, 100V cap in this range, 820μF being the largest. The size of this capacitor is consistent with the ratings.

The main switching transistors are both labelled IRFP4227. The output bobbin is wound on a substantial toroid (35mm diameter) using 1.2mm enamelled copper wire. This amplifier incorporates a speaker protection circuit with a substantial relay. It is more of a high-power AC relay, rated at 30A, but the DC voltage rating is only 30V. Still, we would rather have this in the circuit than not!

#2 Large Switch-Mode Supply

1000W LLC Soft-Switching Power Supply (see Photo 2):
pemag.au/link/abif
pemag.au/link/abig

Claims

- Output power: 1000W
- Input voltage: 220V AC (nominal)
- Output voltage options: ±24V, ±36V, ±48V, ±60V, ±70V or ±80V (±70V in our case)
- Efficiency: 88-93.7%
- Standby power: 2W
- Size: 156 × 100 × 50mm
- Net weight: 350g

Photo 2: this inexpensive "1000W" switch-mode power supply seems to use reasonable quality components and, as far as we can tell, is sufficiently safe. We were pleased that it passed a 500V insulation breakdown test.

It has four input filter capacitors rated at 180μF and 400V, which should provide sufficient headroom at 220-240V AC. The CapXon brand capacitors have a ripple current rating of 700mA each. When delivering 1kW, the ripple current will be just over that. So their ratings are marginal if we use this to its full rated capacity.

The output capacitors are labelled Nichicon 1000μF 80V. These are low-impedance capacitors made for switch-mode power supplies that are the right size and look OK. The mains rectifier is a GBK2510, rated at 1000V & 25A. The output diodes are MURF2040CT 20A ultrafast rectifiers.

The mains-side switching transistors have their part numbers ground off!

As shown in Photo 3, the clearance on this module between Neutral and the mounting screw (which will be Earthed via the chassis) is just over the minimum allowable. However, it is better than the other one we bought and considered (see "The also-ran modules" panel overleaf), so it is OK.

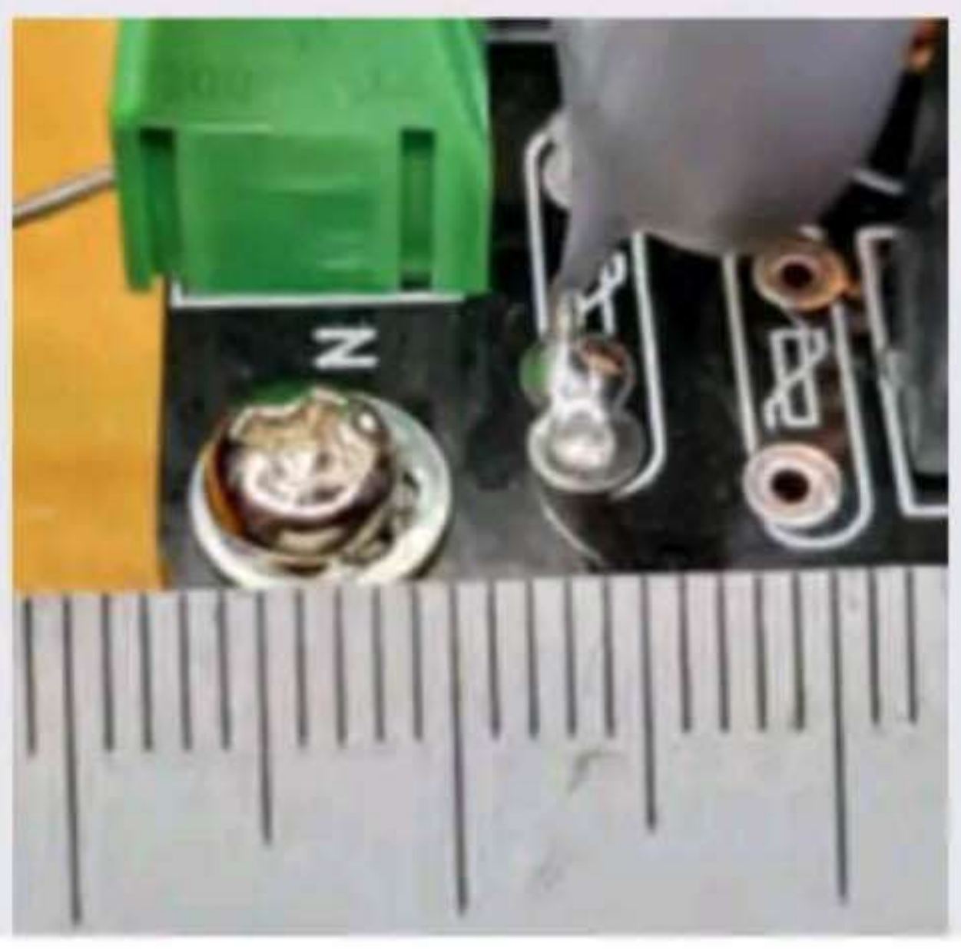


Photo 3: the distance between this component lead that connects to the incoming mains Neutral and the mounting hole is smaller than we would prefer, but is just enough to meet separation standards if Neutral & Active are swapped. That is more common than you might think, especially in old houses.

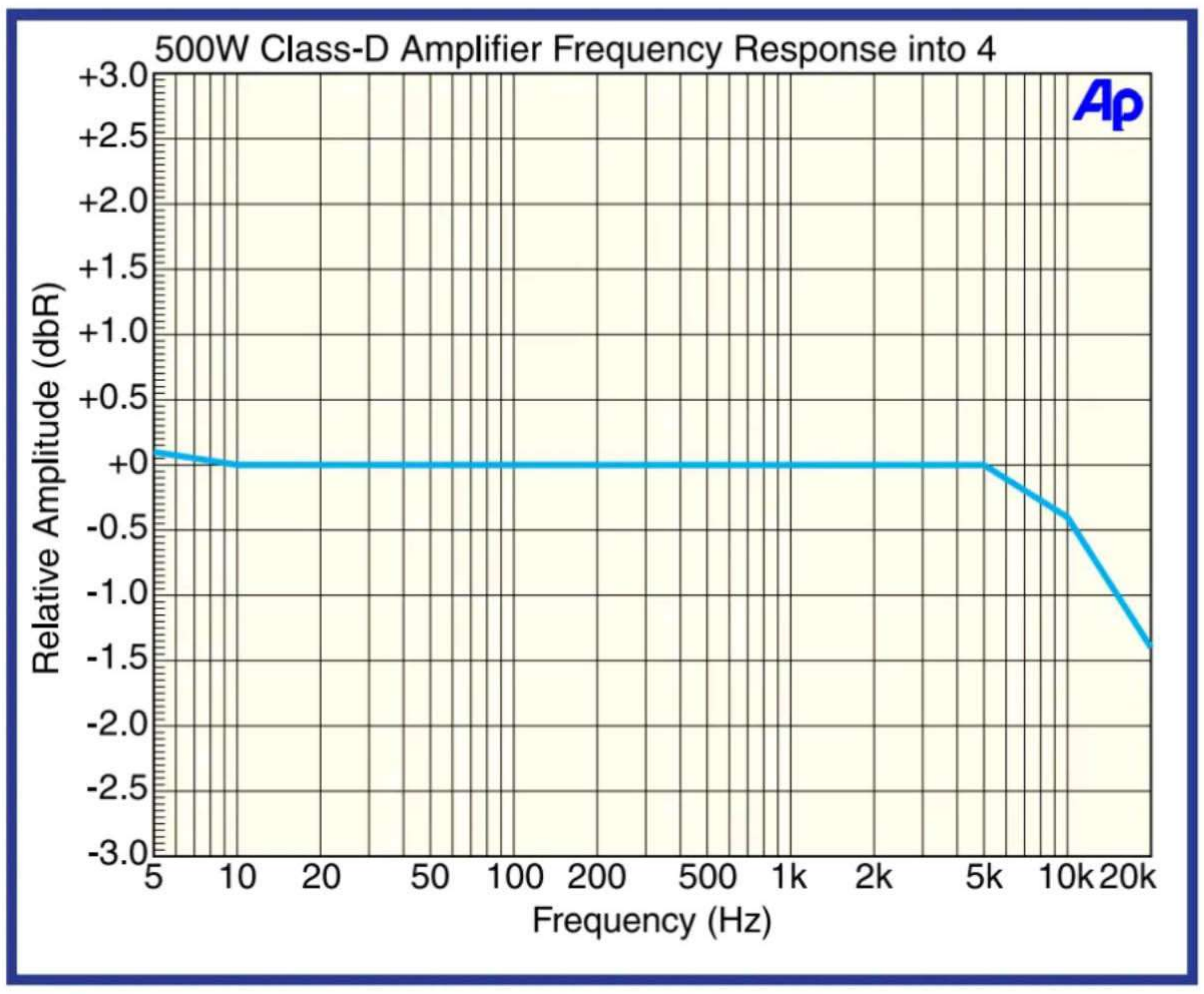
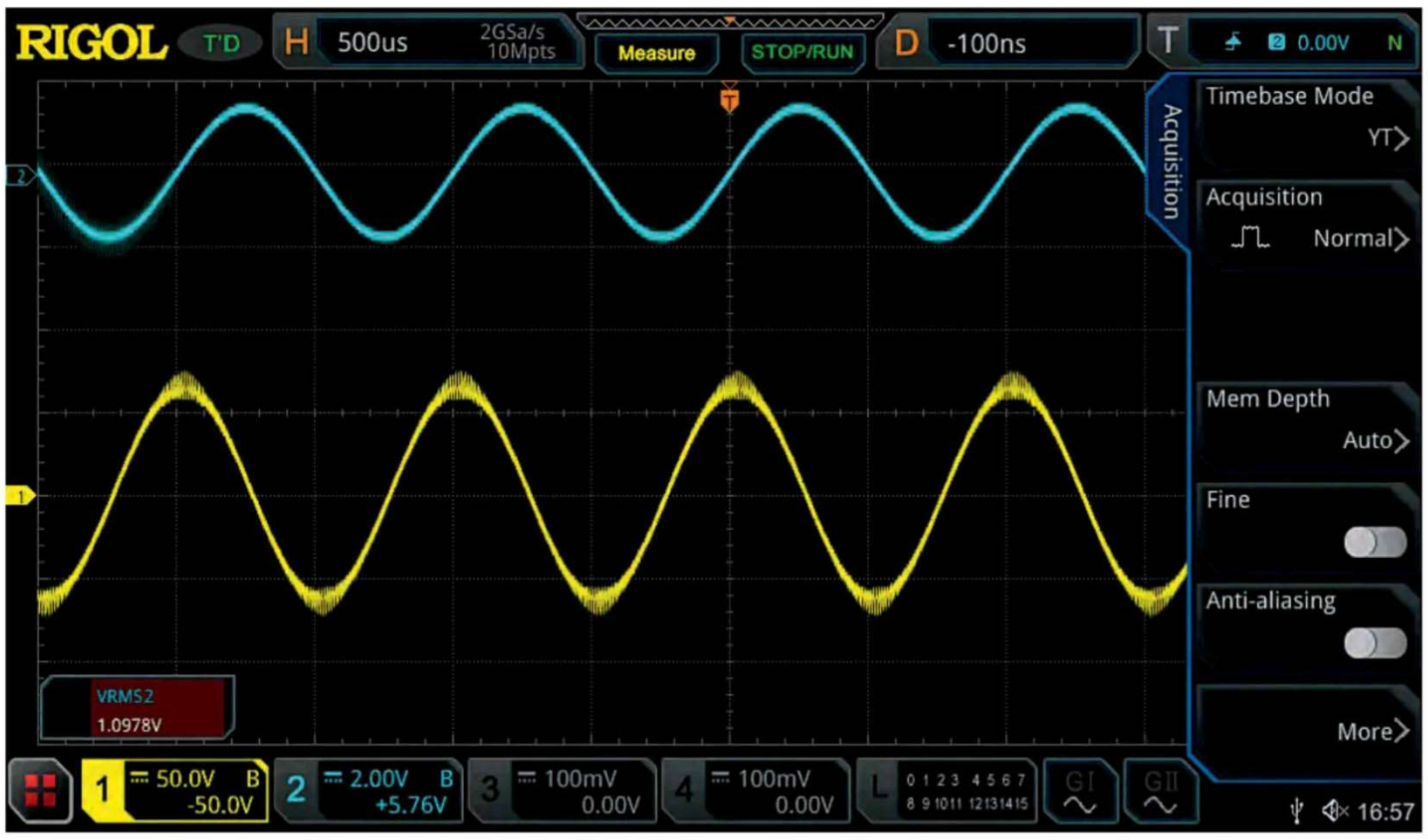
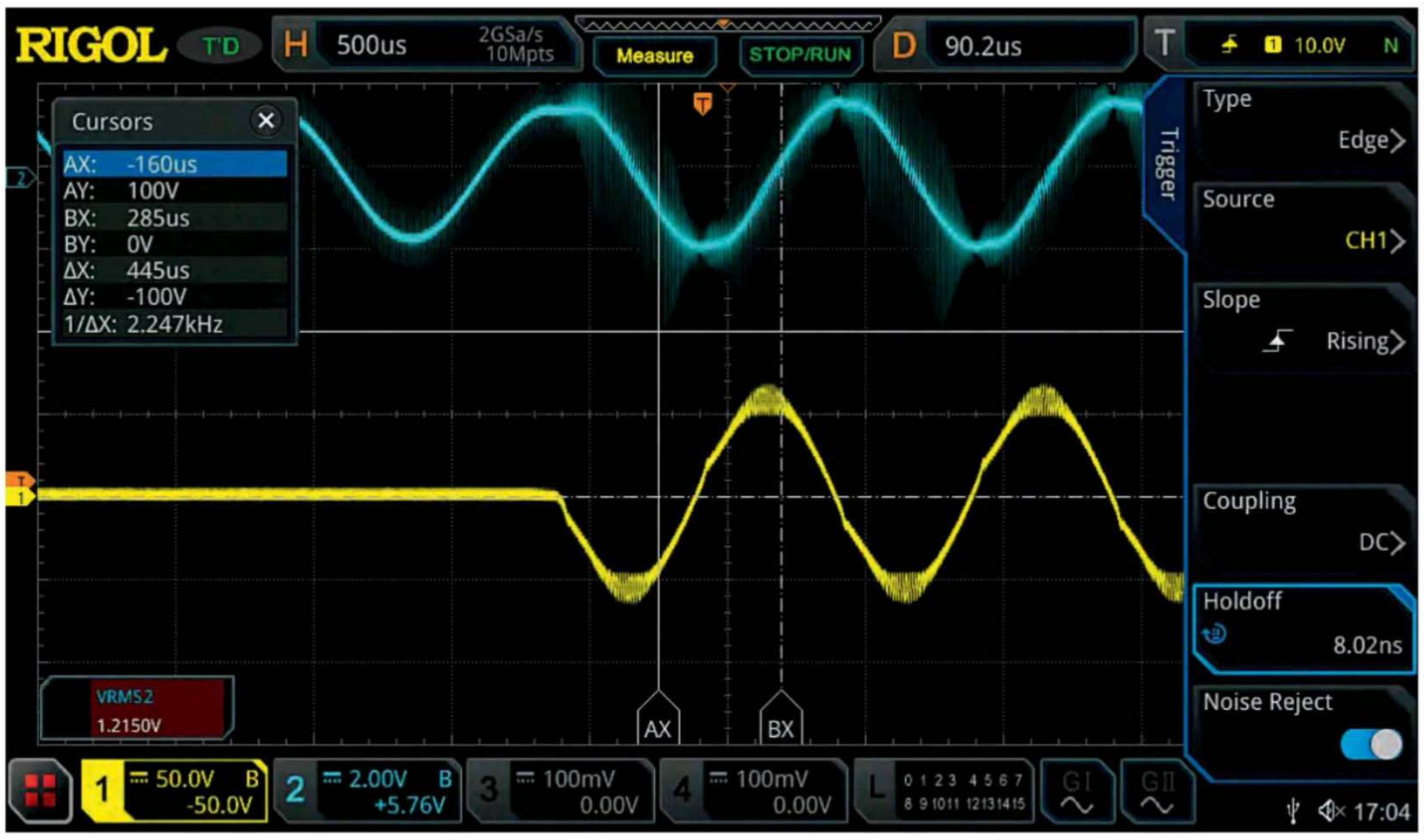


Fig.1: the frequency response of the 500W Class-D module is very flat, dropping by only 0.4dB at 10kHz and 1.4dB by 20kHz. It's definitely suitable for driving an LFE (low-frequency effects) channel, given that there is no such roll-off at the low end.



Timebase Mode YT
Acquisition Normal
Mem Depth Auto
Fine
Anti-aliasing
VRMS2 1.0978V
1 50.0V B -50.0V
2 2.00V B +5.76V
3 100mV 0.00V
4 100mV 0.00V

Scope 1: the amplifier output (yellow) into a 4Ω load near clipping, close to 500W. As it approaches clipping, the Class-D switching frequency drops from 225kHz to about 56kHz, allowing it to deliver a lot of power with some distortion. The 'choppy' appearance of the waveform is normal for Class-D.



Cursors
AX: -160us
AY: 100V
BX: 285us
BY: 0V
ΔX: 445us
ΔY: -100V
1/ΔX: 2.247kHz

Type Edge
Source CH1
Slope Rising
Coupling DC
Holdoff 8.02ns
Noise Reject
VRMS2 1.2150V
1 50.0V B -50.0V
2 2.00V B +5.76V
3 100mV 0.00V
4 100mV 0.00V

Scope 2: the amplifier pulsed output at around 1kW peak into a 2Ω load. There's something nasty going on near the zero-crossings that would lead to very high distortion (if you can see it on a 'scope, it's bad!). Still, it is capable of driving 2Ω as long as the signal dynamic range is high enough.

made by the module suppliers. A distortion level of around 0.02% at 1kHz is not exactly hifi, but it isn't terrible either. It is certainly acceptable for many tasks, especially PA, sound reinforcement, or driving a subwoofer in a hifi or home theatre system.

The frequency response into a 4Ω load is shown in Fig.1. There is a bit of a drop-off at the upper end, but it isn't terrible. It is, however, totally flat down to 10Hz, making it perfect for driving a subwoofer. The slight rise at 5Hz is irrelevant as it is minimal. LFE (low-frequency effects) channel content might go down to 3Hz, at which point it will still be very close to 0dB.

Maximum power testing

Using a 1kHz waveform, the amplifier ran for an extended period delivering 500W into a 4Ω resistive load. When loaded, the 15V rail voltage increases, almost certainly a result of this rail being an unregulated winding on the switch-mode transformer.

The dummy load was a set of 1Ω resistors made from very heavy duty Nichrome wire. At full load, they were just short of red hot, and the heat generated was enough to make it uncomfortable to hold your hand 20cm above the dummy load. The amplifier sustained this on a continuous basis throughout a 20 minute test – see Scope 1.

Reducing the output to about 30V RMS and the load to 2Ω, the protection relay immediately switched off. Assuming this was overload protection, we switched to using a pulsed signal that is more typical of music, with six cycles at 1000Hz followed by 100 cycles of silence and then it repeats.

The amplifier was able to generate this waveform at clipping into 2Ω. The output voltage was about 60V peak, consistent with a claim of close to 1kW – noting that they specify 10% distortion and the tests here were below clipping. The fact that the amplifier shut down for continuous duty but was capable of brief bursts of output is important.

We doubt this amplifier would drive a 2Ω subwoofer with modern music, which can have significant content at low frequencies. The amplifier was happy with a continuous waveform into 4Ω, though. The distortion into 2Ω was visible on the scope (see Scope

The ‘also-ran’ modules

We considered other amplifier & power supply modules when designing this amplifier. The following modules looked OK, but we decided they were not as good as the ones we went with. Some readers might still be interested in using them in different scenarios, although note that the safety of the alternative switch-mode supply is concerning.

#3 Small Class-D amplifier

IRS2092S 1000W Mono Digital Amplifier (see Photo 4)
pemag.au/link/abih

Photo 4: we also tested this Class-D amplifier module which could deliver a similar amount of power. We didn’t choose this one because we’d be running it right at the upper limit of its specified voltage range, whereas the other module has another 10V of headroom and also seems a bit better designed.

Claims

- Supply voltage: ±58V to ±70V
- Output power: 1000W (±70V power supply, 2Ω load, 10% THD)
- Efficiency: ≥90%
- SNR: 90dB
- THD+N (±70V, 2Ω): 1% @ 900W, 0.1% @ 750W
- Frequency response: 20Hz ~ 20KHz
- Speaker load impedance: 2-8Ω
- Voltage gain: 36 times
- Input Sensitivity: 1.5V RMS
- Protection: output short circuit, speaker protection (no relay, though!), over-temperature
- Dimensions: 132 × 68 × 45mm
- Weight: 260g

The output filter capacitors are two 470μF 100V units per rail, labelled Fulkon CD288H. Data sheets were not obvious on the internet, but they look about the right size for the job. The main switching transistors are both labelled IRFP4227, but the labelling is quite different between them. The output bobbin is wound on an E-core using Litz wire, which is reassuring.

#4 Small Switch-Mode Power Supply

LLC Soft-Switching 1000W Power Supply (see Photo 5)
https://pemag.au/link/abii

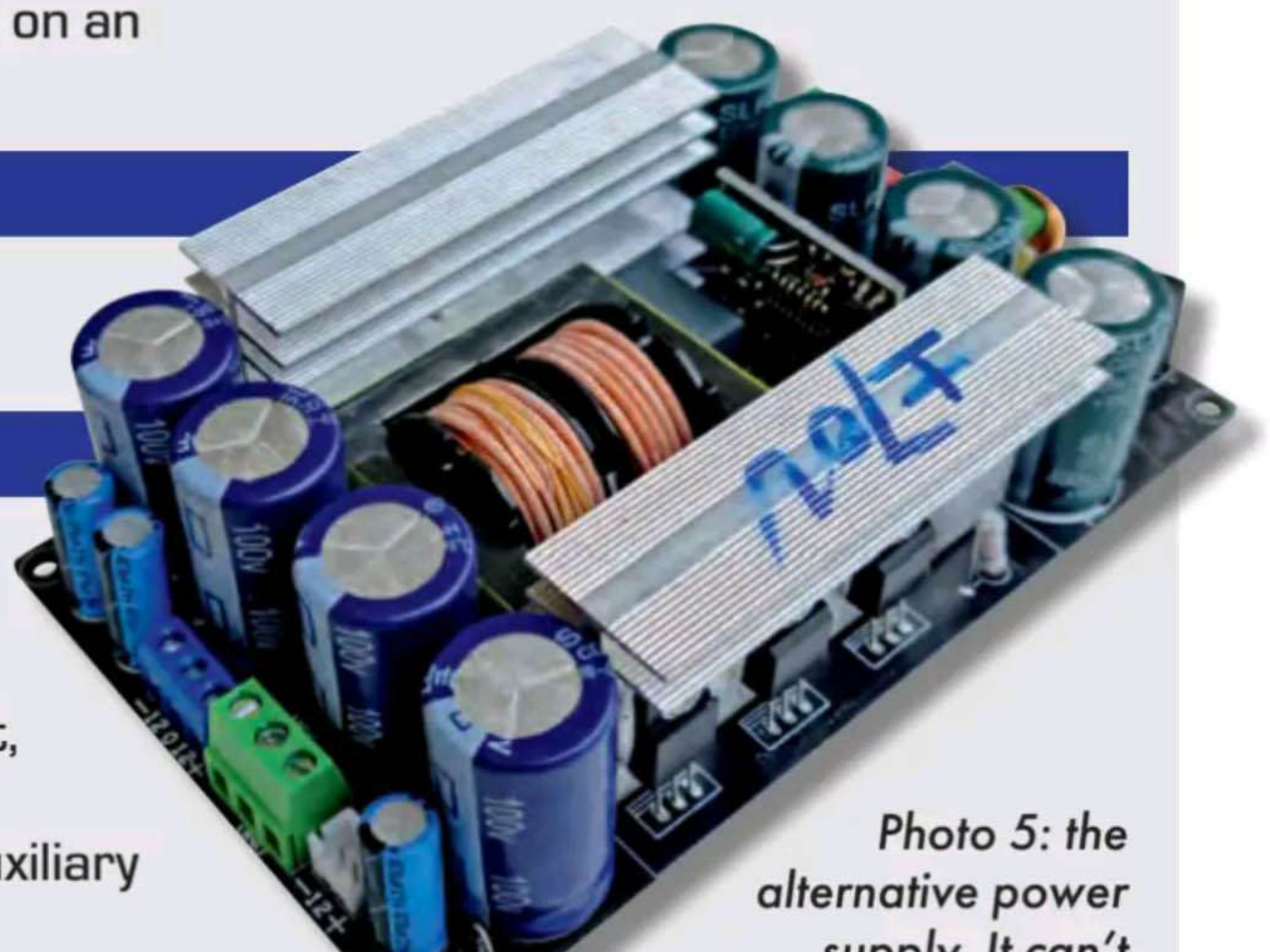


Photo 5: the alternative power supply. It can’t deliver quite as much continuous power as the one we ended up using and seemed to use inferior components that are operated too close to their ratings for our liking (in some cases, beyond!).

Claims

- Input voltage: 200-240V AC
- Output voltage: ±35 to ±80V (±70V in our case)
- Other output voltages: independent 12V, auxiliary ±12V
- Voltage regulation: main ±3% with no load or ±10% with load; independent, ±15% with no load
- Output current/power: 880W for main, 0.5A each for independent and auxiliary
- Continuous power: 500W @ 25°C
- Rated power: 880W for about 5 minutes at 25°C. A cooling fan should be added for long-term operation.
- Peak power: 1200W (less than 100ms)
- Efficiency: up to 95%
- Weight: 400g

There are four input filter capacitors rated at 120μF and 400V, sufficient for running this from 220-240V AC with headroom. The input capacitors are smaller both physically and in capacitance than the preferred unit. At 1kW, their ripple current will be more than 800mA. The data sheet on the installed parts does not specify this parameter, but looking at similar parts, this will likely exceed their rating.

The output capacitors are labelled SLF 1000μF, 100V in the CD288H range, specified for high-frequency and low-impedance. These look right for the job. The mains rectifier is a KBL608 unit rated at 800V, 6A unit. That is marginal.

Somewhat disconcertingly, the clearance from the mounting hole (to an Earthed standoff) and Neutral on this PCB is closer than desirable – see Photo 6. With a shakeproof washer, it is a touch over 2.5mm, right on the edge of acceptability. A solution might be to use no washer or a smaller washer.

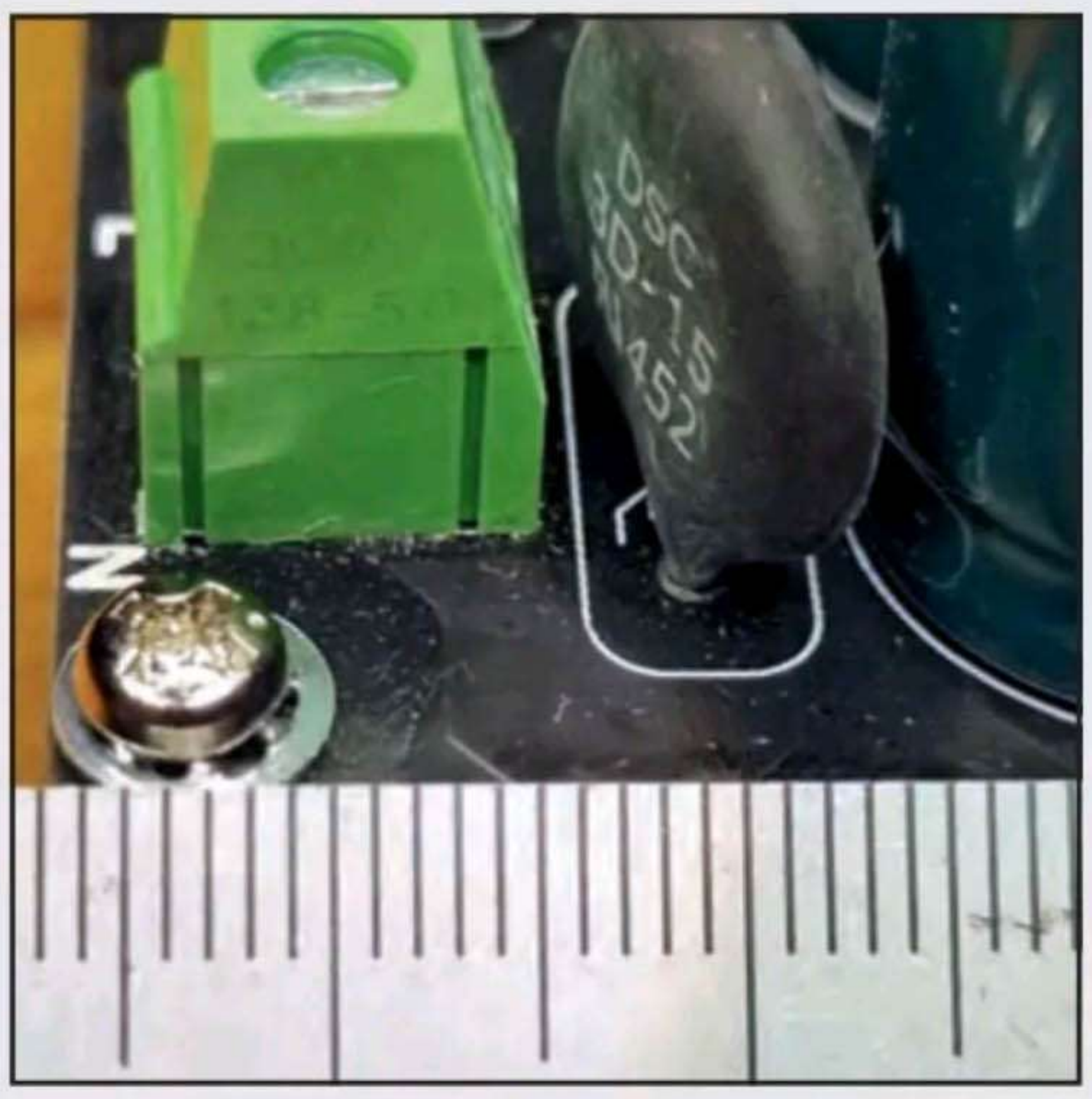


Photo 6: the power supply shown in Photo 5 also has too little clearance between the Earthed mounting hole and the nearest Neutral conductor.

2), so we would dread to think of the actual distortion level.

So in summary, the amplifier ‘does what it says on the box’ aside from delivering that kilowatt into 2Ω.

Design

So, let’s look at what it takes to turn these into a very powerful amplifier. The basic arrangement is shown in Fig.2. It is very much about the appropriate connection of the modules and the provision of some cooling.

This is a ‘monoblock’ amplifier with no volume control. We expect you would feed it from a preamplifier that provides volume control, input switching etc. For stereo use, you would need to build two of these, although if you want to power a subwoofer, one should be fine by itself.

In terms of a preamp as part of a stereo system, you could use our Digital Preamp with Tone Controls from September & October 2022 or our Ultra Low Distortion Preamplifier with Tone Controls from April & May 2020.

You could, in theory, add a volume/level control pot on the front panel and route the signal wiring to the amplifier module via that pot. We’ll leave that as an exercise for our readers as we expect most constructors will use a separate preamp.

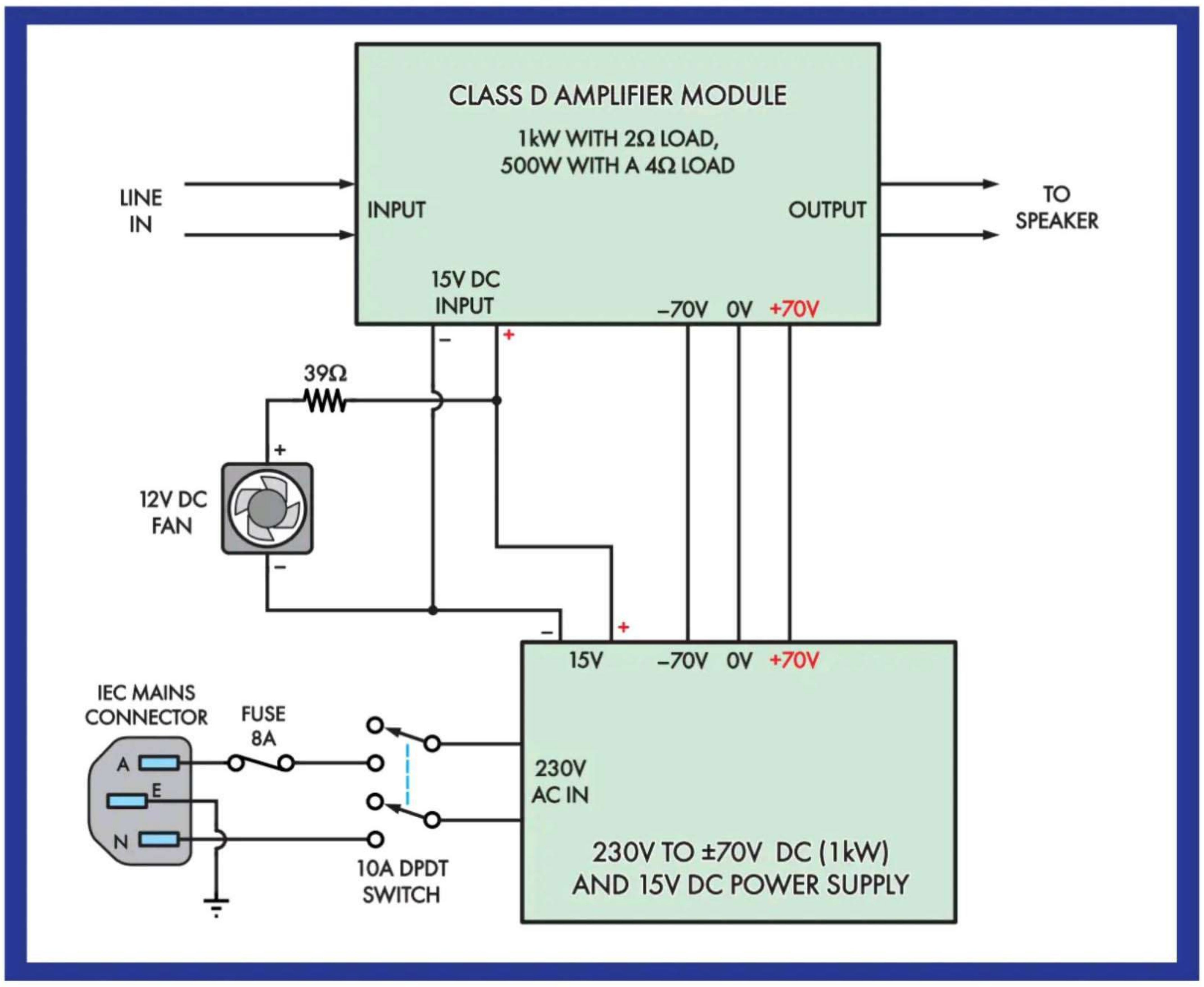


Fig.2: thanks to the prebuilt modules, the ‘circuit’ of this amplifier is dead simple. The power supply generates three rails: -70V, +70V and +15V, which are fed to the amplifier module. The 15V rail also powers the 12V fan via a 39Ω 1W dropper resistor.

Build and testing

We first had to work out how to house this safely and at a reasonable cost. We chose the Jaycar HB5556 chassis as it is just right in size, of good build quality and at a great price. This case also lent itself to us implementing some forced air cooling.

There are three main baffles to keep things cool, as shown in Fig.3.

We are striving to achieve forced airflow over the heatsinks for the power supply and Class-D amplifier. Even though these are better than 90% efficient, if you are driving 1000W into a load, that is 50-100W being dissipated in each module, mainly via their heatsinks. They will get very hot running this way without air moving over them.

Of course, this will not normally be the case. Typical music has a crest factor over 10dB (depending heavily on the type of music), which means that on average, with full-range music not being driven heavily into clipping, the output power would rarely be over 100W for very long.

But consider the realistic use case for a 500W (or 1000W) amplifier; its niche is in subwoofer duty, where, with modern music, all bets are off. Modern music has periods of close-to-continuous bass output. So keeping everything cool is essential.

With modest output, say, averaging up to 100W or so, these amplifier modules are fine in a case with passive cooling. If that is your application, you can avoid manufacturing the plenum presented here. If you intend to play loud music for extended periods, you need to bolster the cooling.

Our plenum is made from three folded sheets of aluminium and uses the case’s lid as the top. This allows

Speaker power handling

Speaker power ratings are a bit of a vexing topic. Those who were around in the 1980s and 1990s will have seen the outlandish Peak Music Power Output or “PMPO” numbers that ran into the thousands of watts, often from a 10W IC amplifier chip!

At a more pragmatic level, the power rating of a loudspeaker is primarily defined by the capacity of the voice coil to dissipate energy and, at a mechanical level, the excursion limit of the cone.

For example, a tweeter typically has a 25mm coil weighing a small fraction of a gram. Many are rated at 100W or more, but the actual continuous power they can handle is only a couple of watts. They rely on the crossover and the nature of music signals to reduce “100W” to only a few watts seen by the tweeter.

Woofers have a much tougher life. AES2-1984 defines the power handling test. Power handling is measured with pink noise with a 6dB peak-to-RMS ratio. For example, the BEYMA 21LEX1600Nd driver has a 3200W “program power” rating and a 1600W continuous power rating, equating to a 400W RMS sinewave power rating.

Be warned that this amplifier could be very bad for the health of your domestic speakers! We have not yet published a speaker design that can handle 500W continuously. However, we plan to publish one in the near future that uses Celestion drivers; we have tested it up to 1000W!

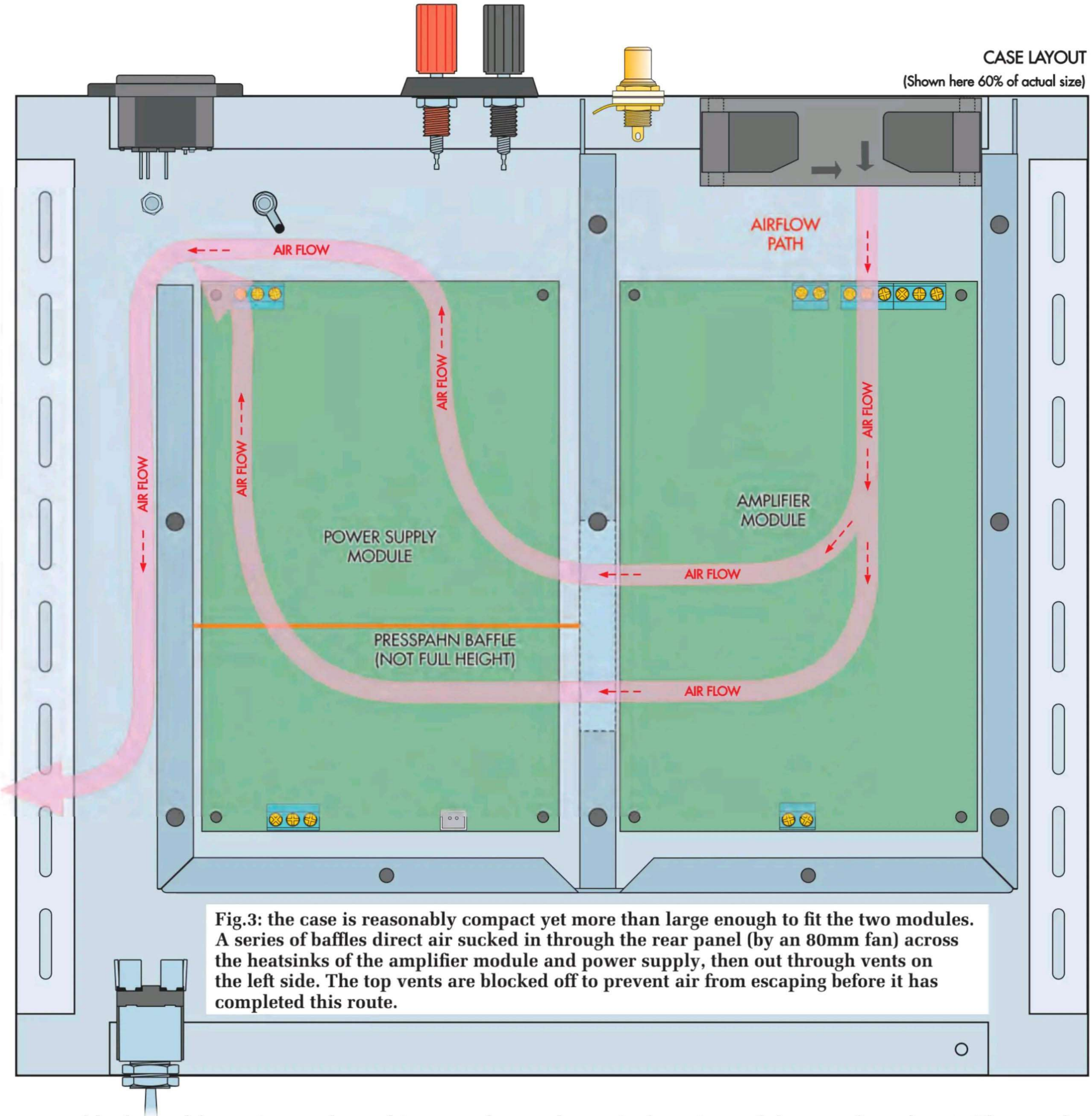


Fig.3: the case is reasonably compact yet more than large enough to fit the two modules. A series of baffles direct air sucked in through the rear panel (by an 80mm fan) across the heatsinks of the amplifier module and power supply, then out through vents on the left side. The top vents are blocked off to prevent air from escaping before it has completed this route.

us to add a fan and force air over the heatsinks, increasing their efficiency.

Without getting too much into the details of heat removal, consider that heatsinks dissipate energy through convection (hot air rising from the heatsink being replaced by cooler air), radiation (mainly IR energy being emitted) and conduction from the heatsink into thermally connected materials. Without running our amp so hot that it's about to melt, radiation is not a significant factor.

Convection is an important means of heat removal, but the case stifles this somewhat, and even in free air, heat will only be removed by convection so fast. By forcing air from outside through the case, over the heatsinks and then exhausting it from the case, we can increase the transfer rate between the heatsinks and the air, picking the heat up off the heatsink and dumping it outside the case.

Making the baffles

We folded aluminium sheets to form a labyrinth, with a fan forcing air in from the rear of the enclosure and using the perforations along the sides of the case for exhaust. The panels are all securely Earthed for safety. The cutting and folding details are shown in Figs.4-6, with instructions to follow. We made ours from three sheets of 1.2mm-thick aluminium, although a thickness between 1.0mm and 1.5mm will be fine.

You could alternatively use polycarbonate sheets and glue or tap and screw them, or if you have the gear, 3D print it. Use our plan as a guide and follow the principles of forcing air across the heatsinks and out of the box.

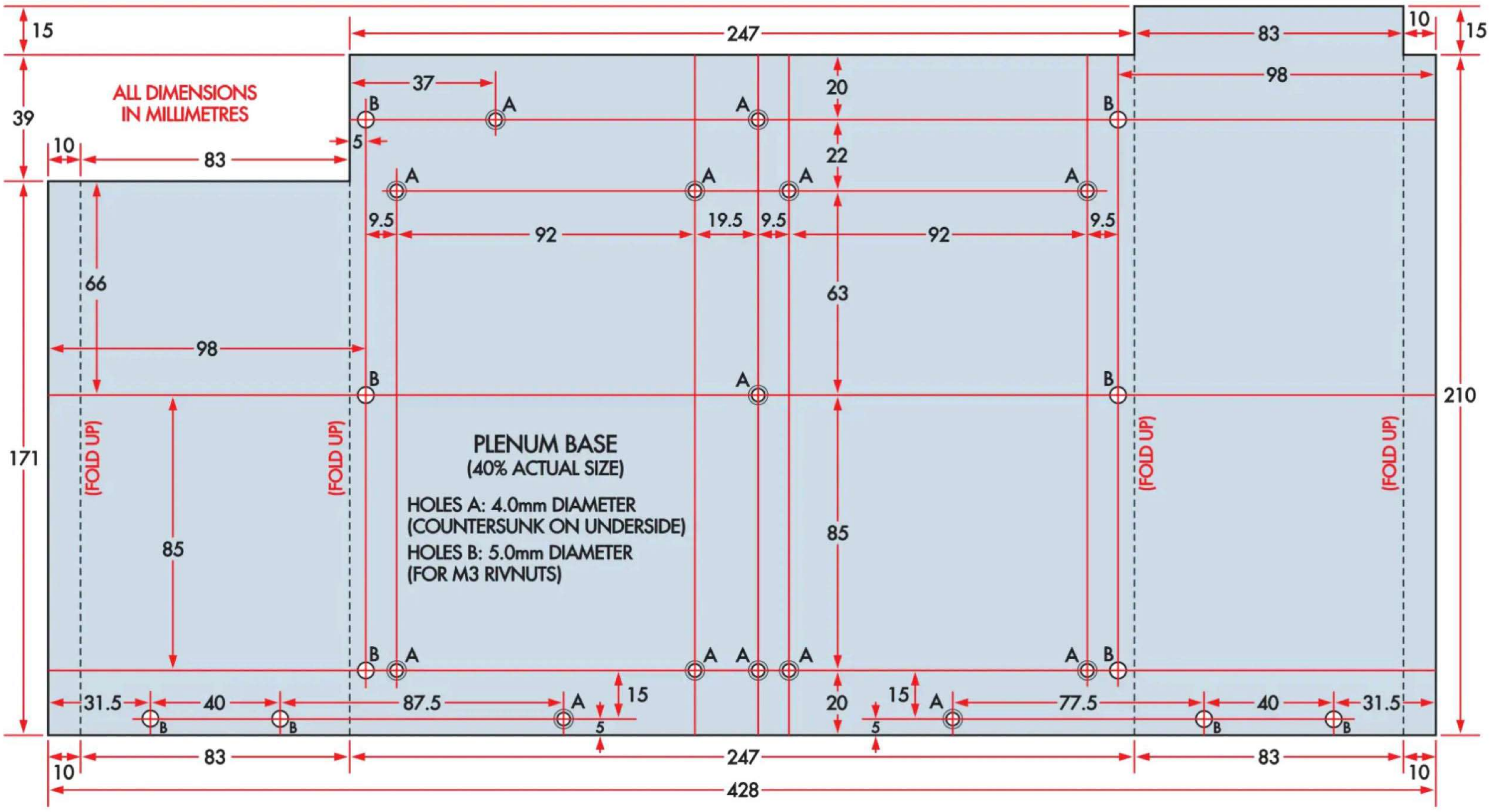

ALL DIMENSIONS IN MILLIMETRES
PLENUM BASE
(40% ACTUAL SIZE)
HOLES A: 4.0mm DIAMETER
(COUNTERSUNK ON UNDERSIDE)
HOLES B: 5.0mm DIAMETER
(FOR M3 RIVNUTS)
(FOLD UP)
(FOLD UP)
(FOLD UP)
(FOLD UP)
A
B
15
39
171
10
83
37
5
247
83
10
15
98
20
22
9.5
92
19.5
9.5
92
9.5
63
66
98
210
85
85
15
31.5
40
87.5
15
77.5
40
31.5
5
20
5
83
247
83
10
10
428

Fig.4: the plenum baseplate is bent up on either side to form the ends of the chamber. The cut-out in the upper left corner is for air to exit into the left-hand side of the case, where it escapes via side vents.

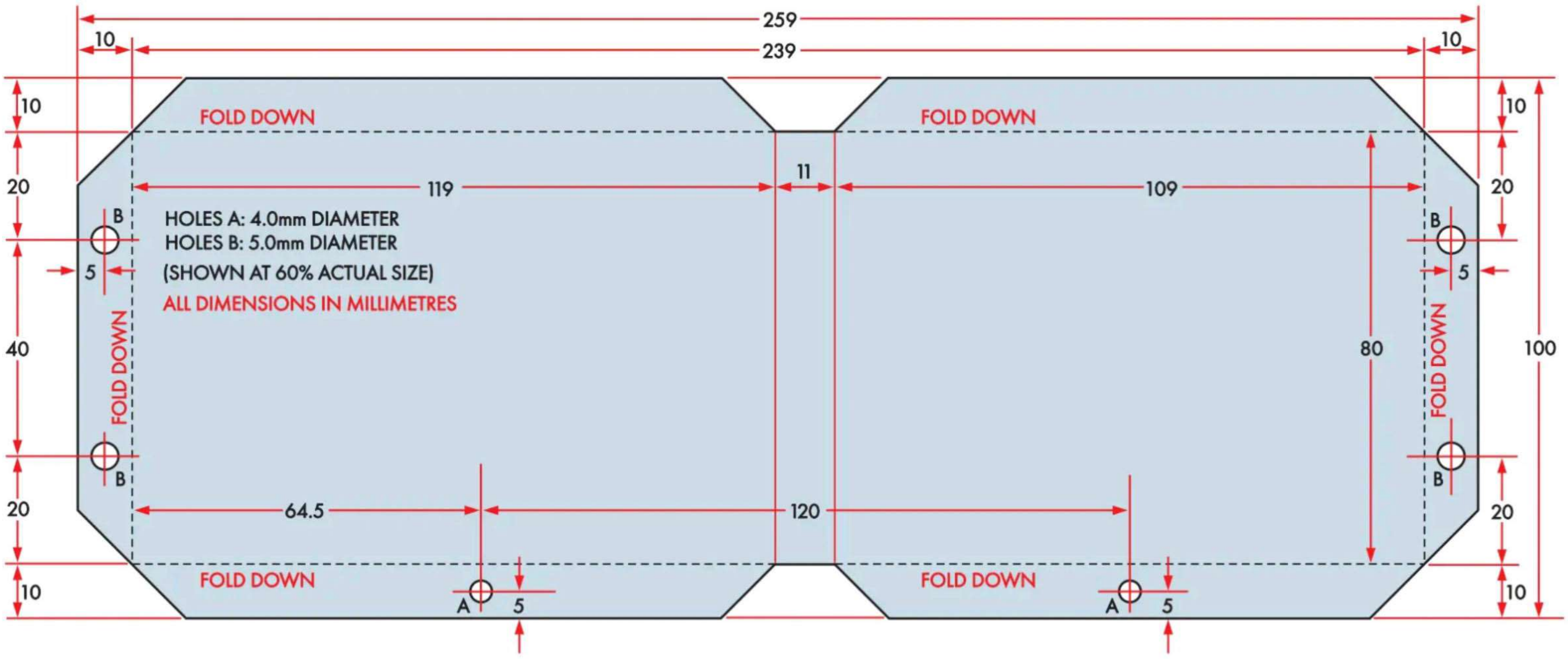

259
239
10
10
FOLD DOWN
FOLD DOWN
FOLD DOWN
FOLD DOWN
FOLD DOWN
FOLD DOWN
HOLES A: 4.0mm DIAMETER
HOLES B: 5.0mm DIAMETER
(SHOWN AT 60% ACTUAL SIZE)
ALL DIMENSIONS IN MILLIMETRES
A
B
10
20
40
20
10
5
119
11
109
80
100
64.5
120
5
5
20
10
20
5
10

Fig.5: this panel, also made from a bent aluminium plate, seals off the section of the plenum chamber closest to the case's front panel.

Assuming you're making the panels as we did, first cut the metal sheets to size. We used a jigsaw. An angle grinder with 1.6mm metal cutting discs also works but requires caution. Drill the holes as shown before bending.

If you do not have a pan brake, 1.2mm aluminium can be successfully bent by clamping it to a workbench with a tight 90° edge and using a hammer and piece of timber to 'panel beat' the corners into the metal sheet. Go slowly and gently as you don't want to work-harden the material.

Make sure the end panel is a good fit for the base. We achieved this by making the base piece first, then, once it was folded, adjusting the folded ends of the rear panel to achieve an acceptable fit. This does not need to be perfect; there will be a fair bit of airflow, so a leak here and there really does not matter.

If you choose to paint your metalwork, make sure to mask off around the Earth lug, as you need a good electrical connection there.

With the baffles made, cut the holes in the rear panel for the fan, input, output and power connectors, as shown in Fig.7. This is an inside view, so if you are cutting from the outside, make sure to mirror it. The final result from the outside (once all the components are mounted) is shown in Fig.8.

Cutting the fan hole is a bit fiddly. We used the 'drill and file' method, in

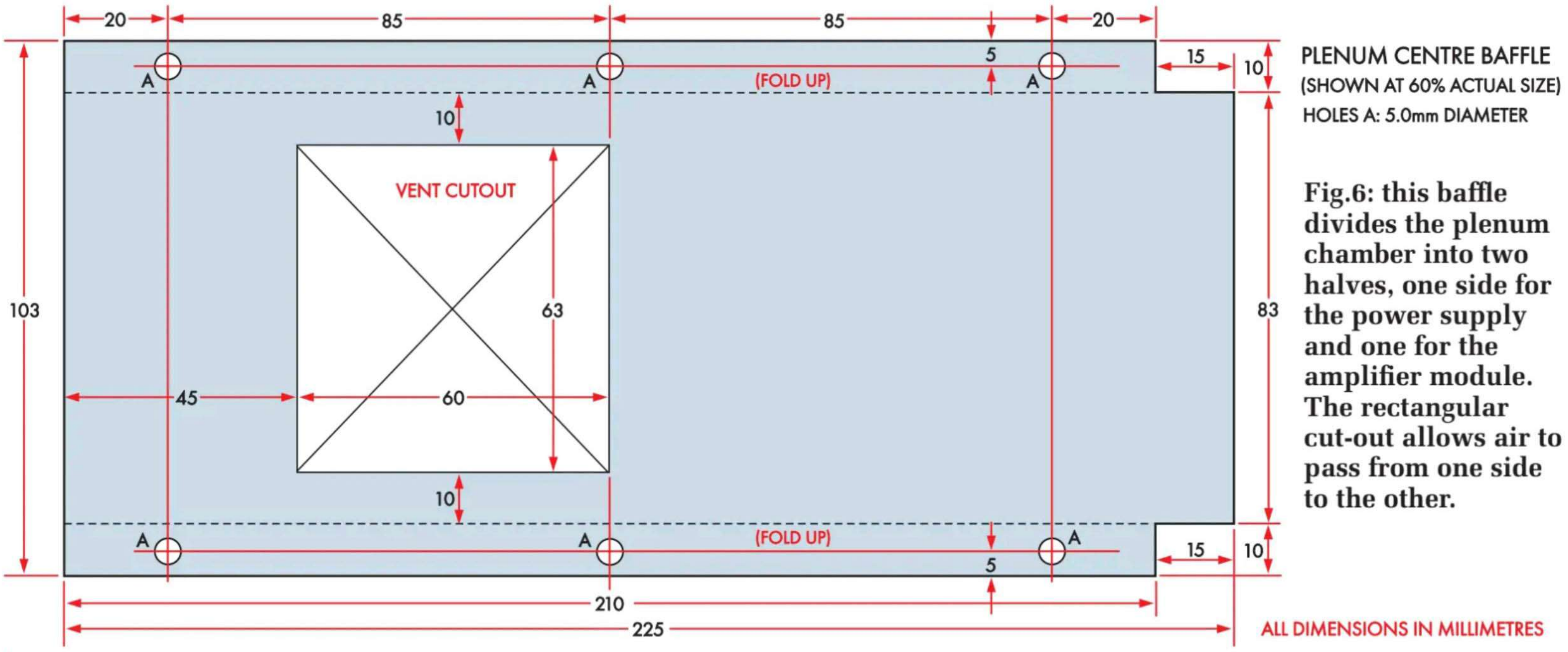

PLENUM CENTRE BAFFLE
(SHOWN AT 60% ACTUAL SIZE)
HOLES A: 5.0mm DIAMETER
20
85
85
20
5
15
10
(FOLD UP)
10
VENT CUTOUT
103
63
83
45
60
10
(FOLD UP)
15
10
5
210
225
A
ALL DIMENSIONS IN MILLIMETRES

Fig.6: this baffle divides the plenum chamber into two halves, one side for the power supply and one for the amplifier module. The rectangular cut-out allows air to pass from one side to the other.

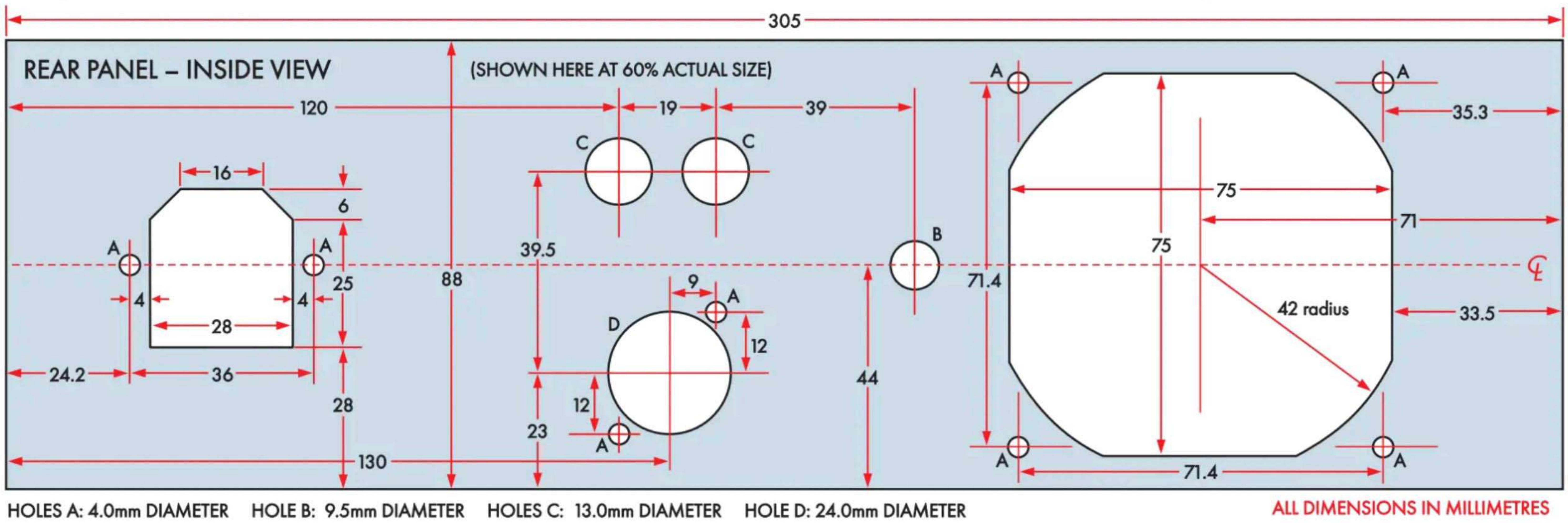

305
REAR PANEL – INSIDE VIEW
(SHOWN HERE AT 60% ACTUAL SIZE)
120
19
39
35.3
C
C
16
6
A
A
25
39.5
88
B
75
71
75
71.4
42 radius
33.5
4
4
28
9
A
D
12
12
24.2
36
28
23
A
44
130
71.4
HOLES A: 4.0mm DIAMETER
HOLE B: 9.5mm DIAMETER
HOLES C: 13.0mm DIAMETER
HOLE D: 24.0mm DIAMETER
ALL DIMENSIONS IN MILLIMETRES

Fig.7: this shows the cut-outs needed in the rear panel but note that the large hole at bottom centre, with two smaller holes near it, is for the Speakon terminal that constructors might opt to leave out. The RCA socket hole has been moved since we built the prototype, as it interfered with the fan.

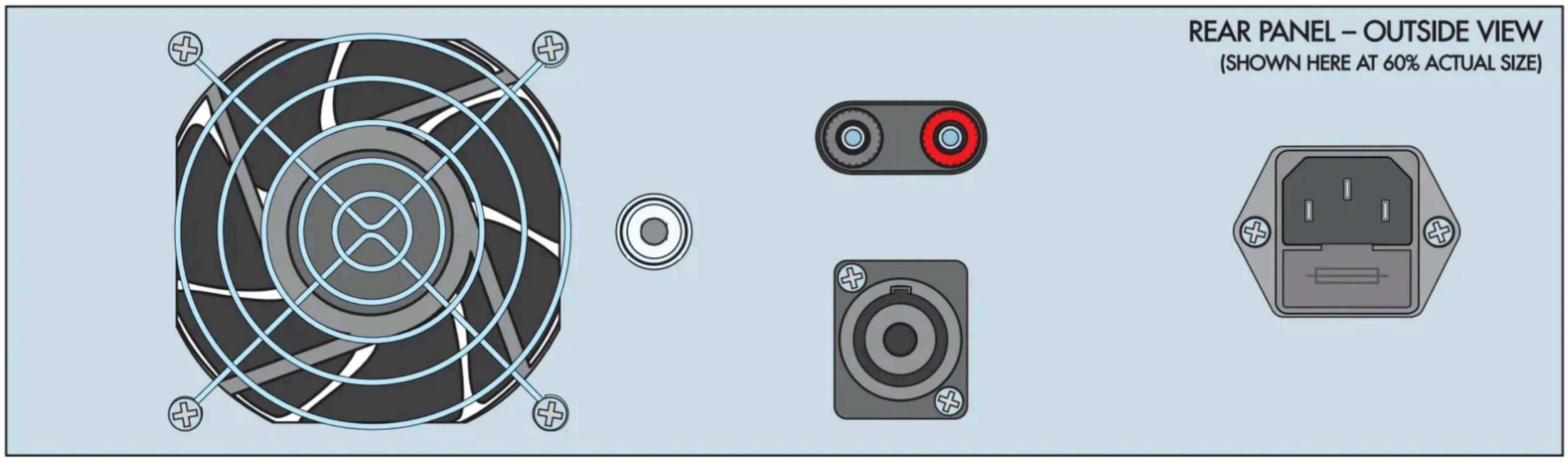

REAR PANEL – OUTSIDE VIEW
(SHOWN HERE AT 60% ACTUAL SIZE)

Fig.8: this shows how the rear panel should look once completed. The Speakon terminal is wired in parallel with the binding posts; only one is required, depending on the speaker connector you plan to use.

which you drill many 4-5mm holes around the inside of the final cut line to remove the bulk of the material, then use a file to smooth the edges. An alternative method is to use a jigsaw with a metal cutting blade.

To make the holes in the base of the case, present the plenum base to the rear panel with the rear panel in the case, then mark the mounting holes. These are shown marked on the drawing; there are six of them between the folds. This will ensure these are in exactly the right spot.

Once marked, drill these, then the mounting holes for the PSU and amplifier modules. These holes need to be countersunk on the underside. Photo 7 shows how we aligned the plenum in the case to drill the mounting holes.

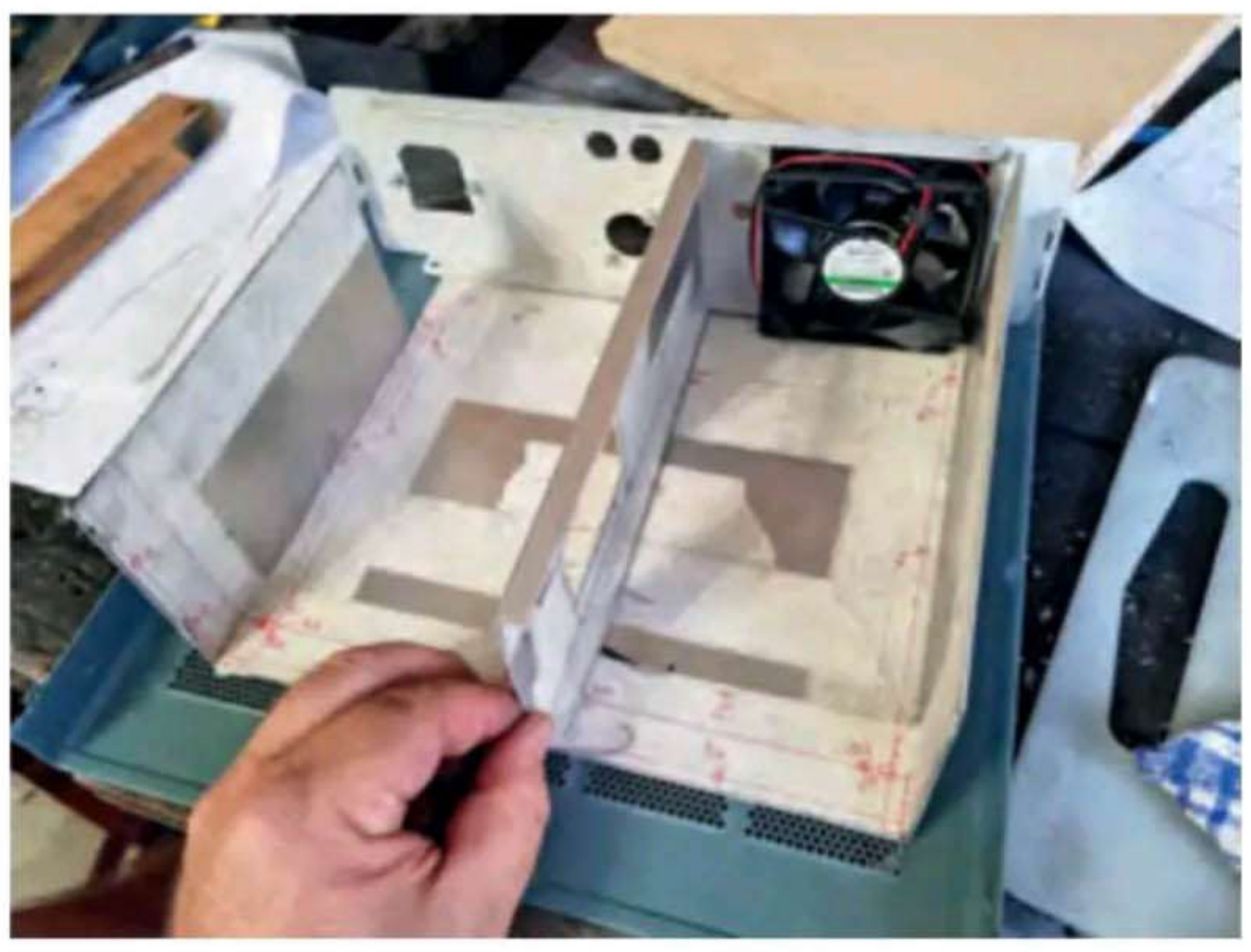


Photo 7: once you've made the plenum base, you can fiddle with the baffle separating the two halves, so it's a good fit and not too much air will leak past.

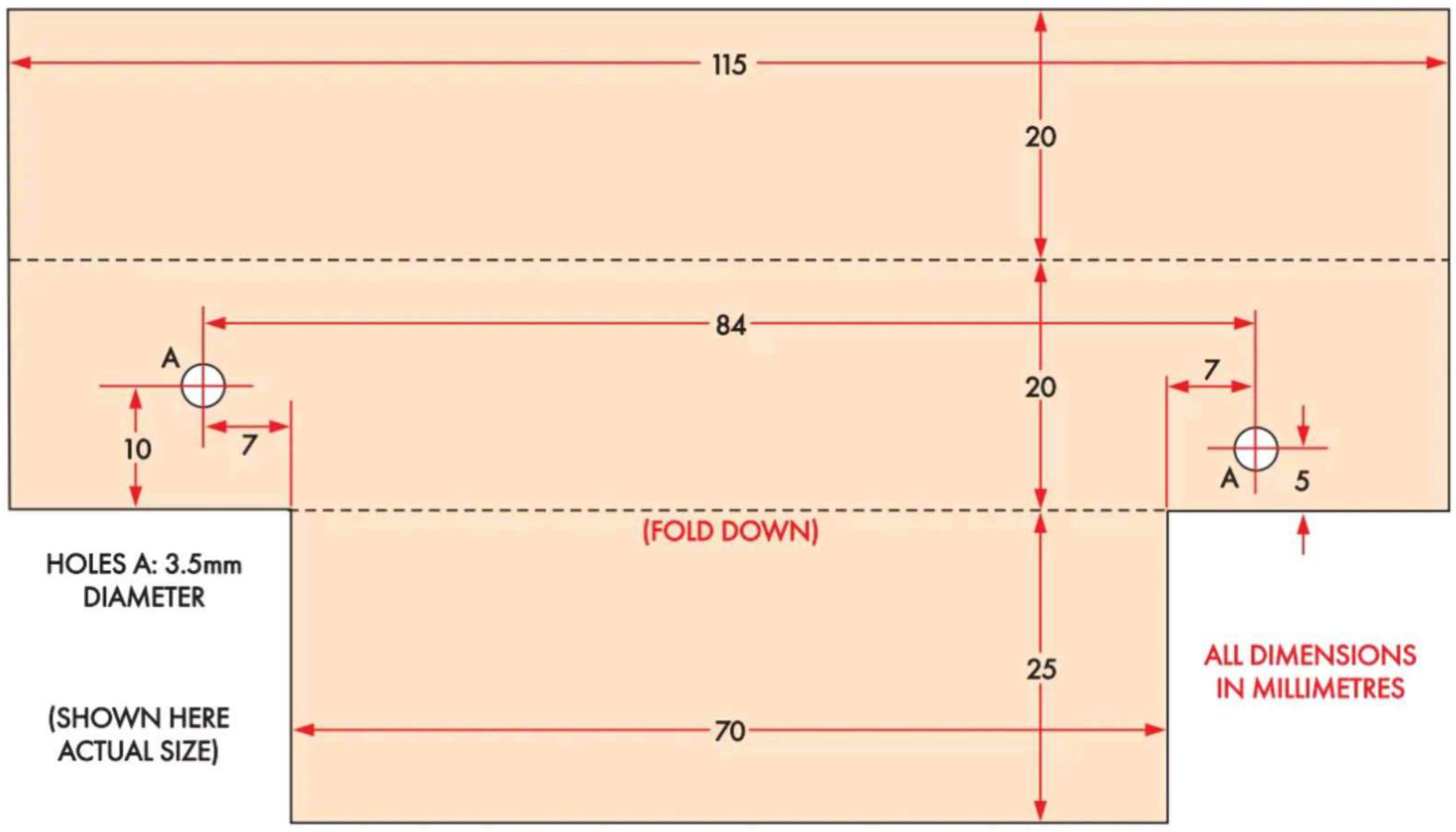


Fig.9: cut a sheet of Presspahn or similar insulating material (thick cardboard will do) and mount it on the power supply to ensure sufficient airflow over both the heatsinks and transformer.

Mount the amplifier and PSU modules now, as shown in Photo 9. Use countersunk M3 machine screws to secure the eight 15mm threaded PCB standoffs to the base. This will allow the plenum assembly to sit flat in the case when assembled.

Then use 6mm M3 machine screws and star shakeproof washers to secure the boards.

Optimising the airflow

We made an extra baffle for the power supply module to force more air over the heatsinks, shown in Fig.9, made from Presspahn. Unfortunately, Presspahn insulating card is becoming hard to get, although we did find an equivalent material (see the parts list).

If you can't get that, use thick cardboard, as we are not relying on its insulating properties too heavily here. Under no circumstances use metal.

This is secured with two M3 machine screws and star shakeproof washers to the tapped holes in the top of the heatsink. Use Loctite to ensure these screws do not come loose over time. You should also stick a piece of card to the inside of the top panel to cover the vent holes over the plenum. This way, the air does not escape through there and has to flow past all the heatsinks on the way out.

Once that's in place, cut and stick lengths of weather-stripping foam along all the top edges of the plenum chamber and baffles, as shown in the photos.

This will make a seal with the case's lid so that too much air doesn't flow

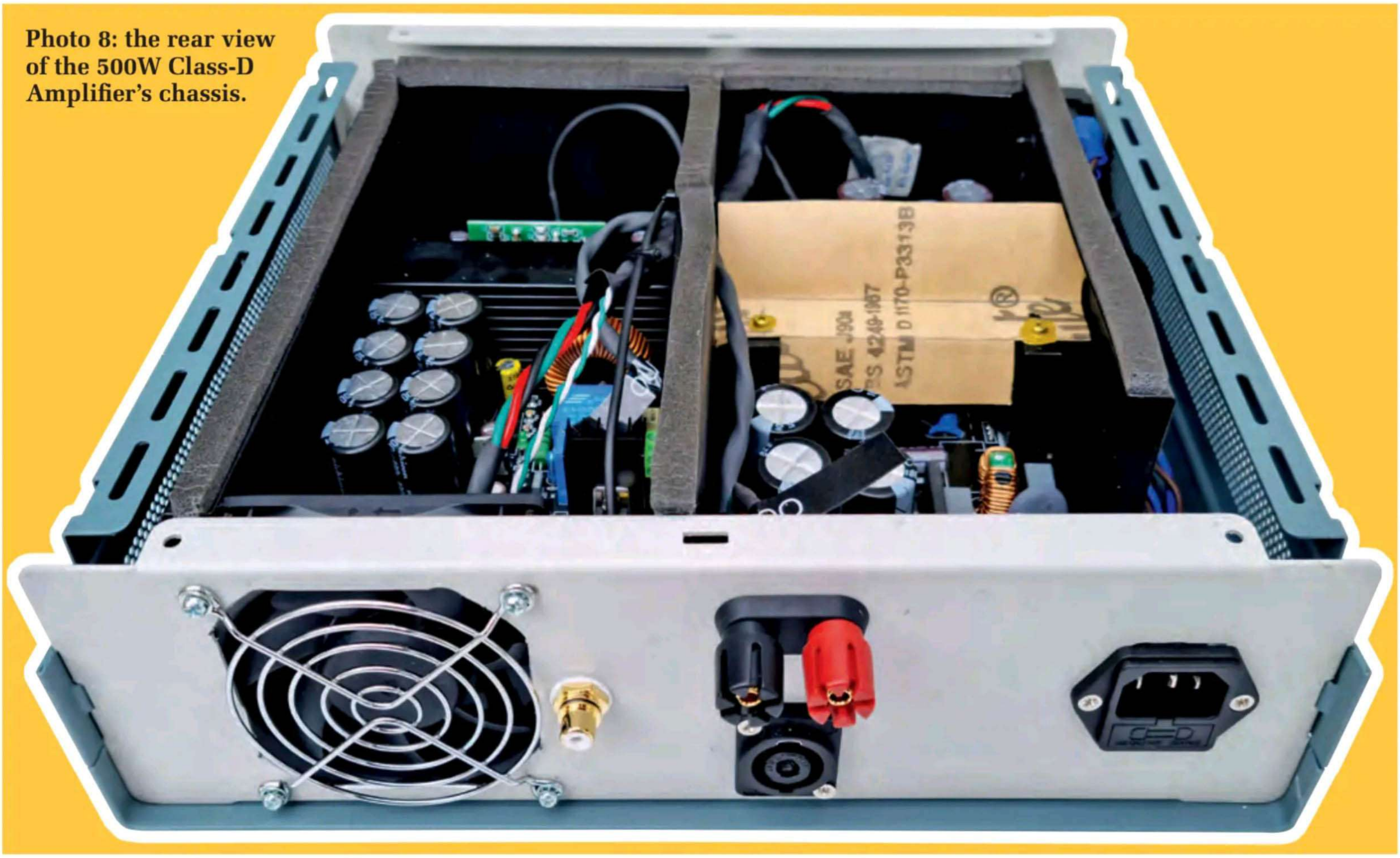


Photo 8: the rear view of the 500W Class-D Amplifier's chassis.

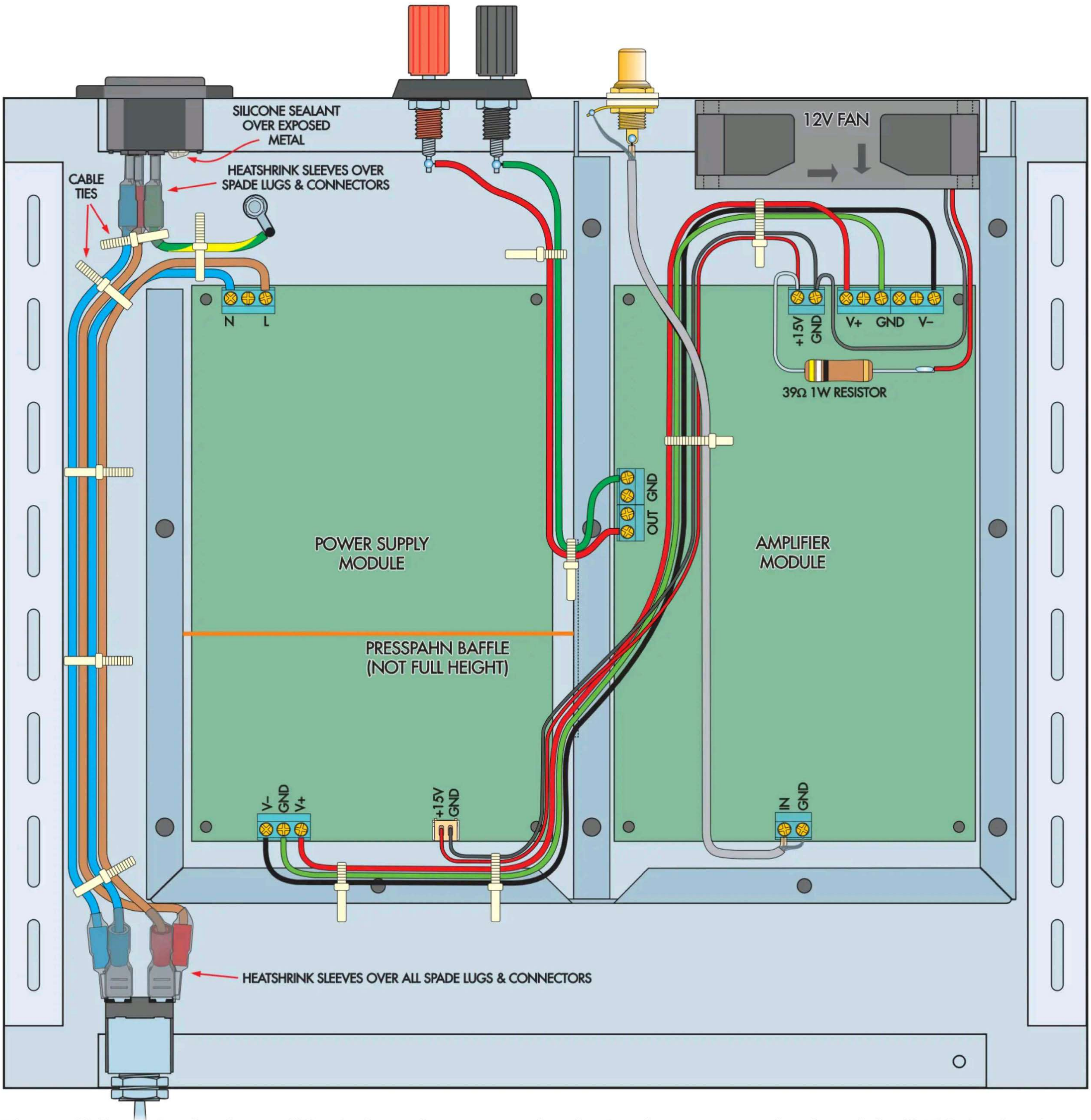

SILICONE SEALANT OVER EXPOSED METAL
HEATSHRINK SLEEVES OVER SPADE LUGS & CONNECTORS
CABLE TIES
12V FAN
N
L
+15V
GND
V+
GND
V–
39Ω 1W RESISTOR
OUT
GND
POWER SUPPLY MODULE
AMPLIFIER MODULE
PRESSPAHN BAFFLE (NOT FULL HEIGHT)
V–
GND
V+
+15V
GND
IN
GND
HEATSHRINK SLEEVES OVER ALL SPADE LUGS & CONNECTORS

Fig.10: all the wiring for the amplifier is shown here, except that the Speakon connector has been left off. If fitting it, wire it in parallel with the binding posts. You could use the spare output terminals on the amplifier module for that if you wanted to. Don't leave off the insulation or cable ties for the mains wiring (also see the photos) and ensure the Earth lug makes good contact with the chassis base.

over the panels and mess up the airflow pattern.

Wiring it up

With the modules installed, mount the internal baffle. This is important as it controls airflow, as shown in Fig.3. You can see how this sits in Photo 8.

Once it is screwed in, install 10A mains-rated red, green and black wire between the ±70V outputs from the PSU to the amplifier module's power inputs, referring to Fig.10. This rating is essential as there is 140V DC between these conductors and they can carry significant current.

Use medium-duty hookup wire to connect the independent 15V power rail to the amplifier module.

Add lengths of 6mm heatshrink tubing over much of these two sets of wires because we will run these cables through the hole in the internal baffle, and we will be tying these to the very top of this opening. This will control where these cables sit, and the heatshrink adds a level of protection and ruggedness to this cabling.

Fan connection

Next, connect the power to the fan. The fan is a 12V type, but the closest rail we have is 15V DC, so a 39Ω 1W resistor connected in series with the fan drops about 3V. The fan is wired

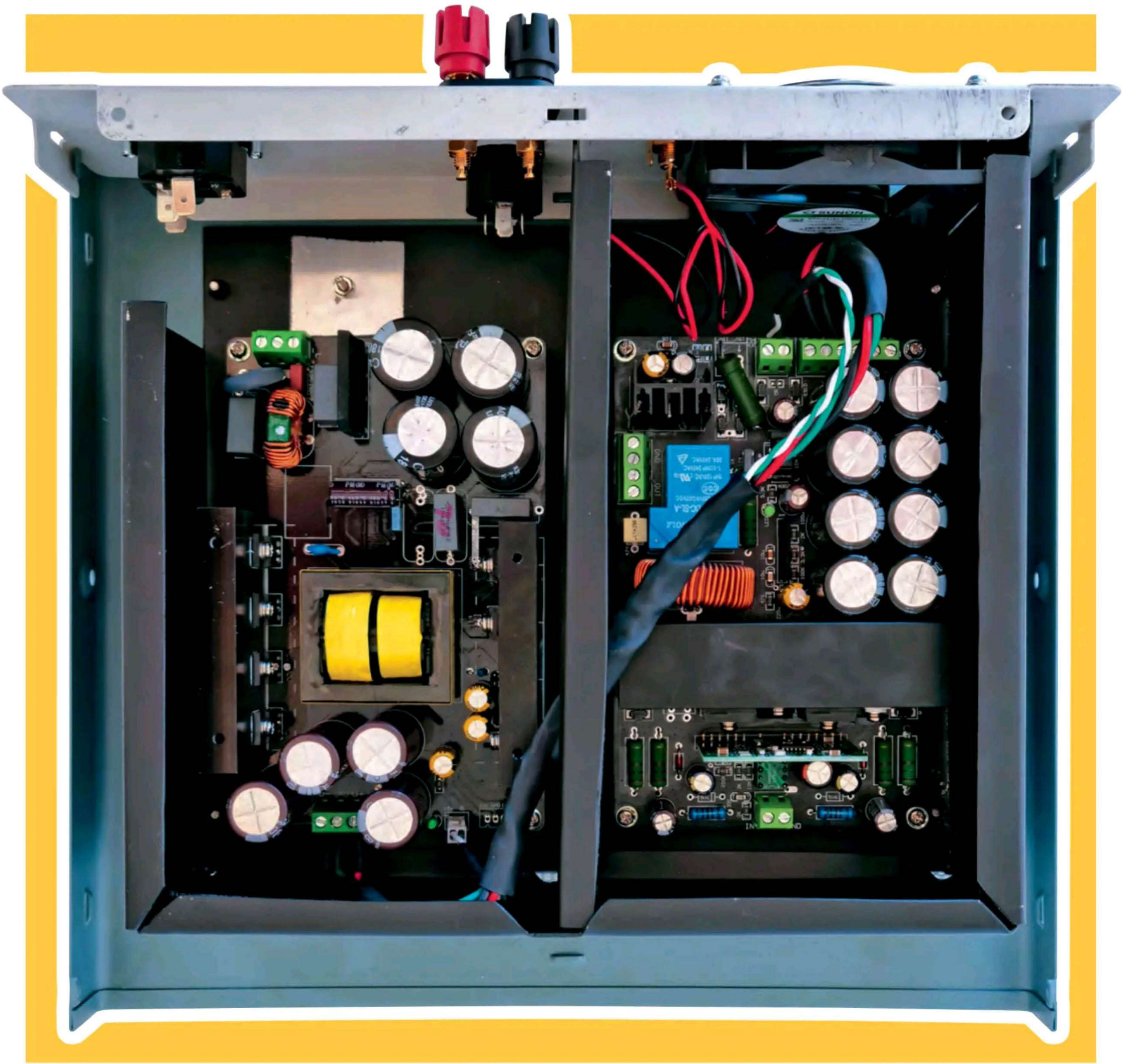


Photo 9: this shows how the two modules fit inside the plenum chamber within the case. The wiring between the two modules has been run along with the input and fan wiring, but the output and mains connections have not been made yet.

to the 15V connector on the amplifier board. Use light-duty or medium-duty hookup wire.

Mains wiring

The mains wiring is also shown in Fig.10. There is not a lot of it; however, you must take caution with all wiring as most is either mains potential or high voltage DC or AC (the output). Ensure all wiring is secured with zip ties to keep it tidy and controlled if anything comes loose.

Install the power switch on the front panel as shown in Fig.11. Then take two lengths of brown and blue mains-rated 10A wire and connect from the IEC mains connector to the switch as shown.

Connect the topmost terminals on the switch to the IEC mains input, and then run a second pair of wires from the central switch terminals back to the mains input on the power supply.

We used insulated crimp connectors on the IEC connector and switch. If you wish to solder these connections instead, insulate the joints with 10mm diameter heatshrink tubing. Keep these wires twisted and tidy, and zip-tie them such that they cannot come loose in the case. We found it handy to label the unswitched and switched input wires.

Using a length of yellow/green striped 10A mains-rated wire, connect the Earth pin on the IEC connector to the M3 Earth screw that runs through the case and plenum metalwork. Before assembling this, take a utility knife and scrape the paint from the case around this bolt. Use a star

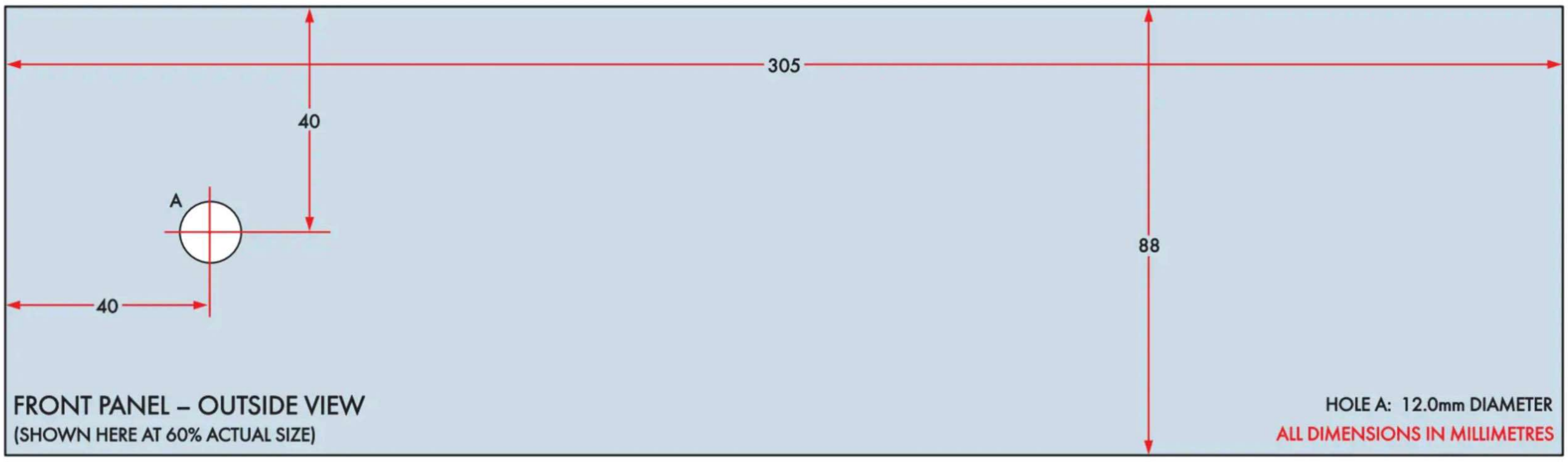

305
40
A
40
88
FRONT PANEL – OUTSIDE VIEW
(SHOWN HERE AT 60% ACTUAL SIZE)
HOLE A: 12.0mm DIAMETER
ALL DIMENSIONS IN MILLIMETRES

Fig.11: just one hole is needed in the front panel for the power toggle switch. That is unless you elect to add a volume control pot or a power-on indicator (an illuminated switch could be used instead).

shakeproof washer on the bottom and top of the case and attach a 3.2mm solder lug to this.

Connect the Earth wiring and check continuity with a multimeter. Install an 8A or 10A ceramic fuse in the IEC mains input/fuseholder assembly. Remember to insulate the exposed metal strip on the back of this connector with neutral-cure silicone sealant, as it will otherwise be live whenever the mains cord is plugged in.

Output wiring

Use mains-rated 10A rated wire for the amplifier output connections. We used 400mm of green and red wire twisted together from the amplifier output to the output connectors. We included both Speakon and binding posts outputs; you may only need the binding posts.

We sleeved the output wiring in a 250mm length of 6mm diameter heatshrink tubing. We did this firstly to ensure there could be no confusion between this and the power wiring and also because we will be securing this to the top of the plenum with a cable tie and want this as extra abrasion protection.

Input wiring

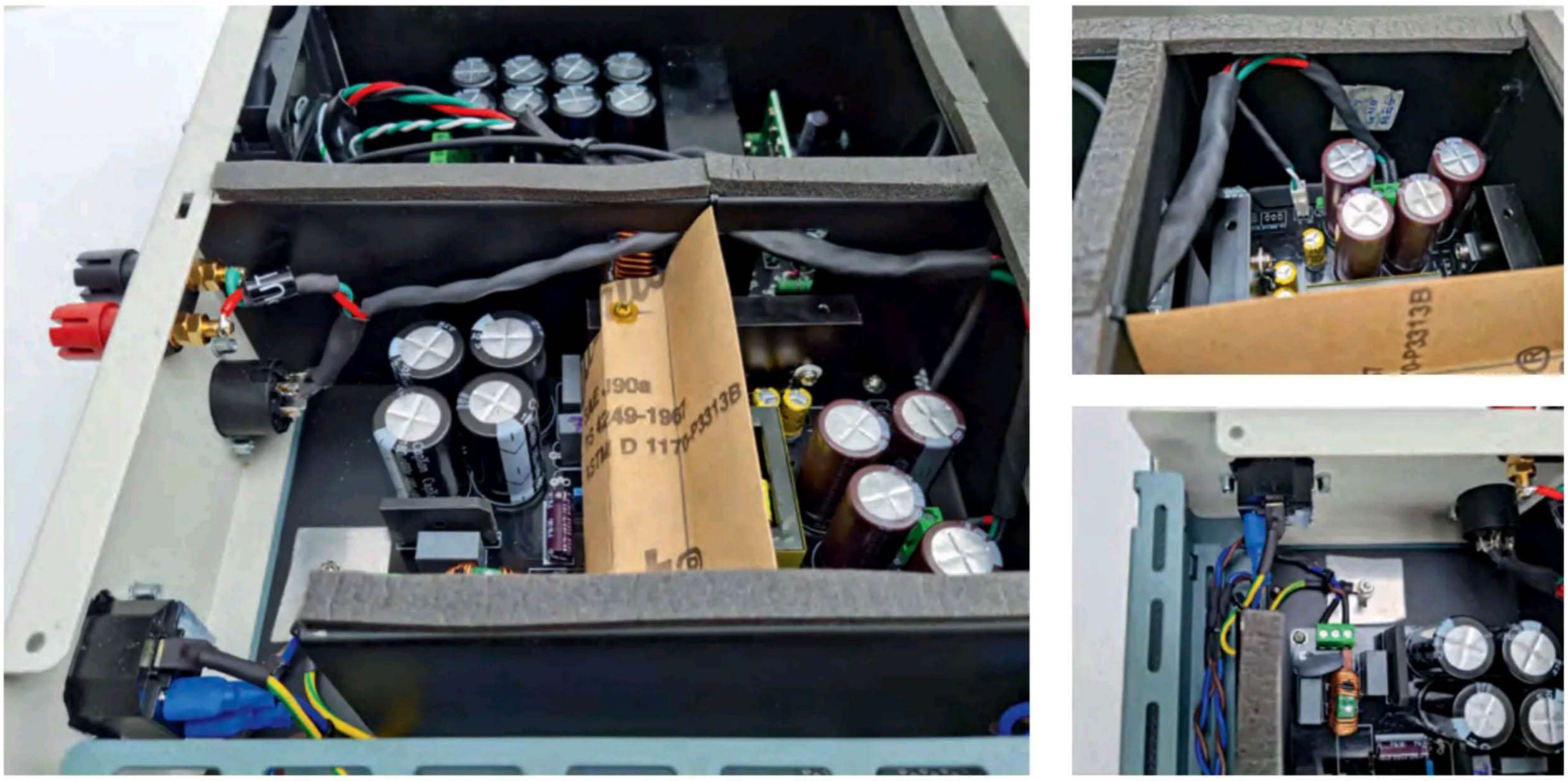
Take 300mm of shielded cable and connect the RCA connector on the rear panel to the screw terminal header on the amplifier board. Use a short length of sleeving to insulate the exposed ground braid and 20mm of 3mm diameter heatshrink to form nice terminations.

Caution

At this point, you should have a standalone chassis with the amplifier modules installed and wired up.

First and foremost: safety. If you are not totally comfortable working with high voltages then do not proceed without help. It's also safest to do the first power up with the lid secured.

This amplifier can generate a lot of power. To do this, it uses high supply rails of ±70V DC. It could easily stop your heart if you make contact with these two rails. Also, the switch-mode power supply operates from the mains



This view shows how we wired up the output connectors and gives you a good view of the Presspahn baffle that optimises airflow over the power supply module. You can also see how the mains input wiring has been insulated. Also note how the output wiring and ±70V rail wiring is cable tied to the top of the plenum, just behind the Presspahn baffle.

and has close to 400V DC in parts of the circuit. This is also lethal.

Second: danger to your possessions. The amplifier generates 44V RMS continuously into a 4Ω load. This is close to 500W. If you feed this into your speakers as a sinewave, we can guarantee you will destroy them. See the panel on "Speaker power handling".

Similar cautions apply for test equipment; make sure that if you connect this to a distortion analyser, it is on a 50V or 100V RMS range.

Testing

First, check that the mains power switch is on, then with it unplugged, do a final check with a DVM on its 20MΩ range (or similar) and check for any measurable resistance between the Active and Neutral inputs and the output ground connector. If there is, then you need to stop and find the problem.

Also perform a final check of your wiring. A fault here will be both spectacular and dangerous.

Plug the amplifier in, switch it on and listen for the speaker protection relay switching in after a couple of seconds.

Carefully measure the voltage between ground, V+ and V– on the power supply output using some properly insulated DMM probes and a suitably rated meter. The rails should both be within 5V of 70V but with different polarities.

Carefully measure the voltage on the +15V input to the amplifier and ensure it is close to expected.

If any of the above fails, unplug the amplifier and leave it off for 10 minutes. After verifying that the mains plug is still out, disconnect the power amplifier from the power supply so you can check the PSU by itself. If you can't see the right voltages at its outputs with no load, you have a faulty PSU.

If the PSU measures OK, rebuild it and check your wiring carefully.

Now plug in a signal generator to the input and a CRO with a 10:1 probe set to measure up to 70V peak to the output. Power up and look for the sinewave on the output. Increase the signal level until you see clipping; check that this is about 40-44V RMS.

Connect a load and start the input signal at a low volume level, increasing to a manageable level. Only use a loudspeaker for this if you have no other choice and are happy to test at moderate levels only. If you have a dummy load, run the amplifier at as high a power as is safe for your load for 5-10 minutes. If you are using a speaker for the test, play some moderately loud music.

At this point, we are really just checking that nothing goes wrong – no puff of magic smoke etc. After testing as hard as you feel safe, unplug everything and open the amplifier. Use an IR thermometer to measure the temperature of the PSU heatsinks, the E-core transformer on the PSU (in our tests, this was the hottest part) and the amplifier heatsink.

If these are all below 65°C, everything is fine and you are all set! Otherwise, check the airflow management components (baffles, seals etc) to verify that there are no massive air leaks and confirm that you haven't skipped any of the steps listed above.

Parts List – 500W Monoblock Amplifier

1 1000W Class-D amplifier module (see links at the start of the article)
1 1000W 70V split-rail switch-mode power supply (see above)
1 vented metal bench enclosure, 87 × 279 × 305mm **[Cheval SAL2.0D2W2 or similar, available from Amazon UK]**
1 dual binding post for speakers
1 panel-mount insulated RCA socket
1 fused IEC mains input socket
1 10A+ mains-rated chassis-mount DPST/DPDT toggle switch
1 8-10A fast blow sand-filled or ceramic M205 fuse
1 Speakon chassis-mount speaker connector (**optional**)
1 quiet 80mm 12V fan
1 80mm fan guard
1 2-way 2.54mm-pitch vertical polarised header
1 2-way 2.54mm-pitch polarised header plug
1 39Ω 1W resistor

Hardware
1 428 × 225 × 1.0-1.5mm aluminium sheet (for base)
1 225 × 103 × 1.0-1.5mm aluminium sheet (for baffle)
1 259 × 100 × 1.0-1.5mm aluminium sheet (for plenum end)
1 115 × 65mm sheet of Presspahn or similar insulating card **[www.ebay.com.au/itm/293254125529]**
9 M3 × 16mm panhead machine screws
12 M3 × 10mm countersunk head machine screws
16 M3 × 6mm panhead machine screws
12 M3 hex nuts
32 M3 star shakeproof washers
8 15mm M3-tapped spacers
1 3.2mm solder lug
9 blue insulated 6.3mm female spade crimp lugs for 1.5-2.5mm^2 wire
1 1.2m length of 9-10mm wide adhesive foam weather stripping
1 1.5m length of 5-10mm wide adhesive foam weather stripping
1 pack of small Nylon cable ties

Wire & cable
1 1.5m length of brown mains-rated 10A hookup wire
1 1.5m length of blue mains-rated 10A hookup wire
1 0.5m length of green/yellow striped mains-rated 10A hookup wire (eg, stripped from a length of 10A three-wire mains flex)
1 1m length of red mains-rated 10A hookup wire
1 1m length of green mains-rated 10A hookup wire
1 1m length of black mains-rated 10A hookup wire
1 0.5m length of red medium-duty hookup wire
1 0.5m length of black medium-duty hookup wire
1 300mm length of single-core shielded audio cable
1 1m length of 6mm diameter clear heatshrink tubing
1 200mm length of 3mm diameter clear heatshrink tubing

Practical Electronics

UK readers SAVE £1 on every issue

SUBSCRIBE NOW!

01202 087631

Take out a one-year subscription and save more than £10 over the year.
Even better – save over £1 per issue if you subscribe for two years – a total saving of £26.

Overseas rates represent exceptional value

You also:
- **Avoid any cover price increase for the duration of your subscription**
- **Get your magazine delivered to your door each month**
- **Guarantee a copy, even if your local newsagent sells out**

Order by:
- **Phone or post with a cheque, postal order or credit card**
- **Or order online at: www.electronpublishing.com**

Practical Electronics

SUBSCRIPTION PRICES

Subscriptions for delivery direct to any address in:

UK: 6-months £33.99, 12-months £59.99, 24-months £114.99

Europe Airmail: 6-months £38.99, 12-months £69.99, 24-months £129.99

Rest Of The World Airmail: 6-months £46.99, 12-months £84.99, 24-months £164.99

Cheques (in **£ sterling only**) payable to *Practical Electronics* and sent to: Practical Electronics Subscriptions, PO Box 6337, Bournemouth BH1 9EH, United Kingdom

Tel: 01202 087631
Email: pesubs@selectps.com

Also via our online shop at**: www.electronpublishing.com**

Subscriptions start with the next available issue. We accept MasterCard or Visa.
(For past issues see the Back Issues page.)

DIGITAL SUBSCRIPTIONS

Electronic subscriptions are available from £29.99 for 12 months, more details at: **www.electronpublishing.com**

SUBSCRIPTION ORDER FORM

☐ 6 Months: UK £33.99, Europe £38.99 (Airmail), Rest of the World £46.99 (Airmail)
☐ 1 Year: UK £59.99, Europe £69.99 (Airmail), Rest of the World £84.99 (Airmail)
☐ 2 Years: UK £114.99, Europe £129.99 (Airmail), Rest of the World £164.99 (Airmail)

To: **Practical Electronics Subscriptions**
PO Box 6337
Bournemouth BH1 9EH
United Kingdom

Tel: 01202 087631
Email: pesubs@selectps.com

I enclose payment of £ ....
(cheque/PO in £ sterling only)
payable to *Practical Electronics*

☐ Please charge my Visa/Mastercard

My card number is: ....
Please print clearly, and check that you have the number correct

Card Security Code .... Valid From Date.......................
(The last 3 digits on or just under the signature strip)

Card Ex. Date ....

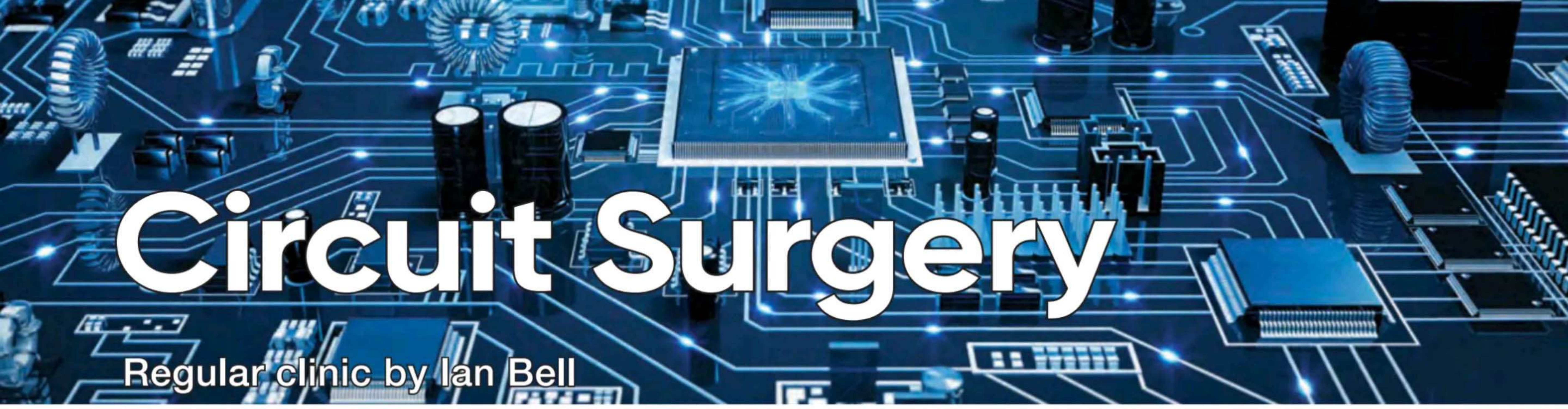
Name ....

Address ....

Post code .... Tel. ....

Email ....

Subscriptions start with the next available issue.



Circuit Surgery

Regular clinic by Ian Bell

Topics in digital signal processing – digital-to-analog converter (DAC) outputs

We are looking at various topics related to digital signal processing (DSP).

DSP covers a wide range of electronics applications where signals are manipulated, analysed, generated, stored or displayed as digital data, but originate from, and/or are output as, real-world signals for interaction with humans or other parts of the physical world.

Fig.1 shows the key elements of a generic DSP system, with a signal path from an analog input via digital processing to an analog output. This does not necessarily represent every DSP system (nor are all the parts necessarily shown), but it serves are as reference for the various subsystems we will look at.

Last month, we discussed the filtering requirements for antialiasing and reconstruction in a DSP system (these blocks are show in Fig.1).

In previous discussions, we have looked at the input sampling process in some detail but have not considered the DAC output and reconstruction process in as much depth, so that will be our focus this month.

Reconstruction filters

As discussed last month, the spectrum of a sampled signal contains copies of the original signal (spectral images) centred on integer multiples of the sampling frequency (f_s), as illustrated in Fig.2.

If the bandwidth of the original signal (B) conforms to the Nyquist criteria, meaning that it is less than half the sample frequency, then the spectral images do not overlap and can, in theory at least, be removed by filtering to obtain the original signal.

In Fig.2, the bandwidth the spectral image centred on f_s extends down to $f_s - B$, which is just higher than $f_s/2$. Fig.2 also shows the response of an ideal 'brick-wall' reconstruction filter with a cutoff frequency of $f_s/2$ that will remove the replica spectra centred on f_s and higher frequencies.

Of course, we cannot have an ideal filter in practice, but if the sampling frequency is above the Nyquist limit, there is some headroom to work with (as illustrated by the gap between $+B$ and $f_s - B$ in Fig.2). However, this does not guarantee that the signal can be perfectly reconstructed with a filter that we could actually build.

Ideally, the filter must completely remove the replica spectra, but the closer the sampling rate is to the Nyquist rate, the more demanding the filtering requirements become.

Fig.3 shows a case where the filter passes the original signal fully but does not completely remove the first spectral image. If a lower cutoff frequency was used to remove more of the image, some of the original signal would be attenuated. In both cases, the signal is not perfectly reconstructed from the samples.

There is a trade-off between the sampling rate and filter requirements. The higher the sampling rate is relative to the original signal bandwidth, the easier the filtering is by increasing the separation from the first spectral image (lower cost, simpler filters). However, higher performance is required from other parts of the system.

DAC output signal

The discussion of the reconstruction process so far has assumed that the sampled waveform is a sequence of scaled impulses (idealised pulses that occur only at the sample time). However, the output from a DAC is not an ideal sampled waveform; it is a sampled-and-held signal, also called a zero-order hold.

In the July 2024 issue, we discussed the sample-and-hold circuits that are commonly found at the inputs to analog-to-digital converters (ADCs), as shown in Fig.1. These are required to capture the analog input signal and then hold a steady value while the ADC processes it.

They typically use a capacitor with a voltage that follows the input until a switch isolates it, whereupon it retains the voltage it had across it at that instant. The DAC output has a similar waveform shape, but is produced by a different mechanism.

The DAC will contain a digital register that holds the value to be delivered. The bits from this register will control electronic switches (transistors), or voltage inputs, to an analog circuit

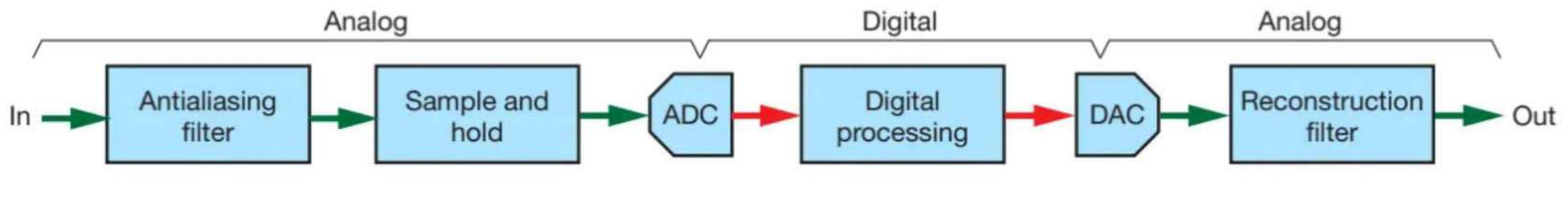


Fig.1: A generic digital signal processing (DSP) system structure.

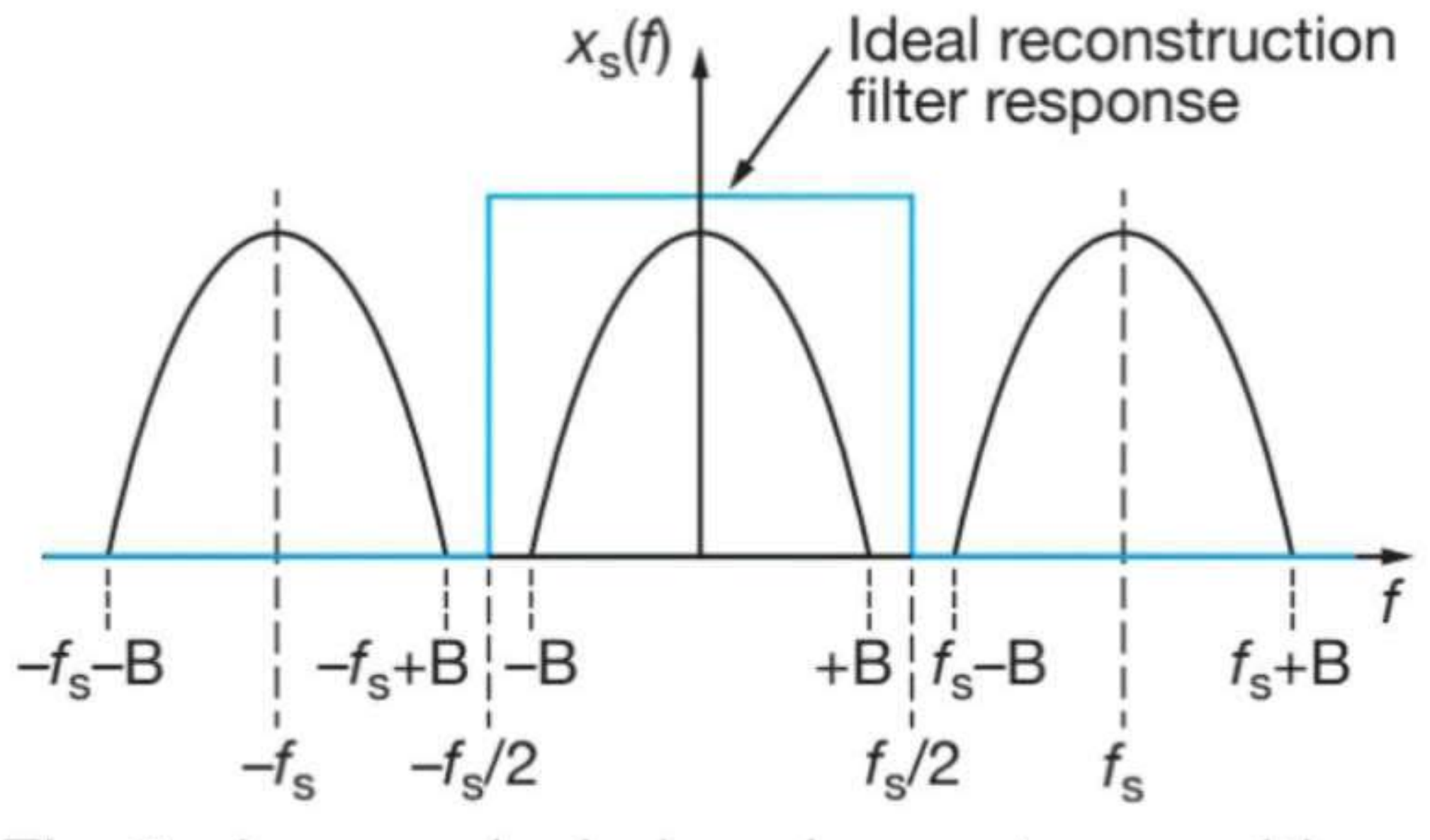


Fig.2: A sampled signal spectrum with an ideal reconstruction filter response.

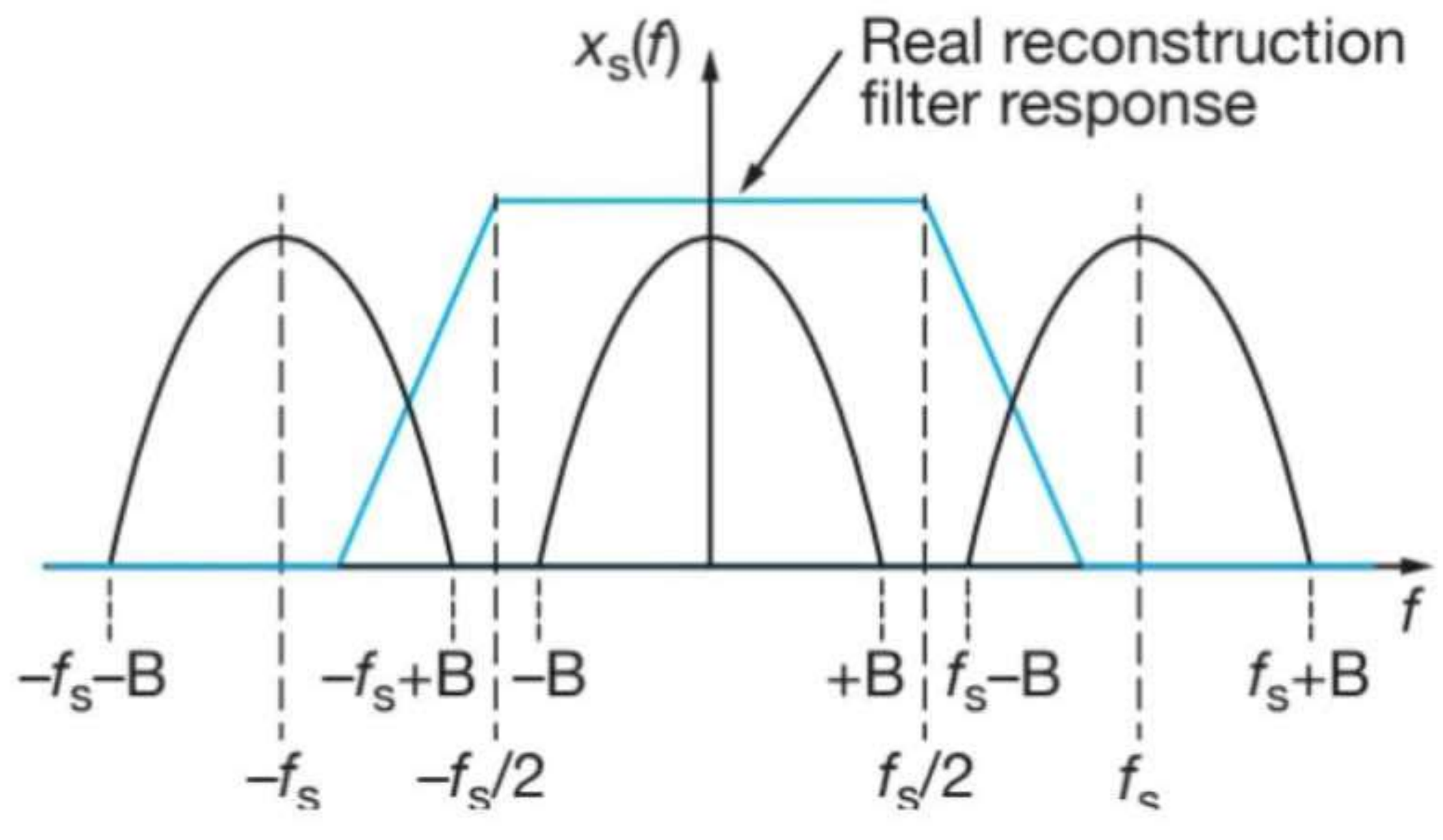


Fig.3: A sampled signal spectrum with a realistic reconstruction filter response.

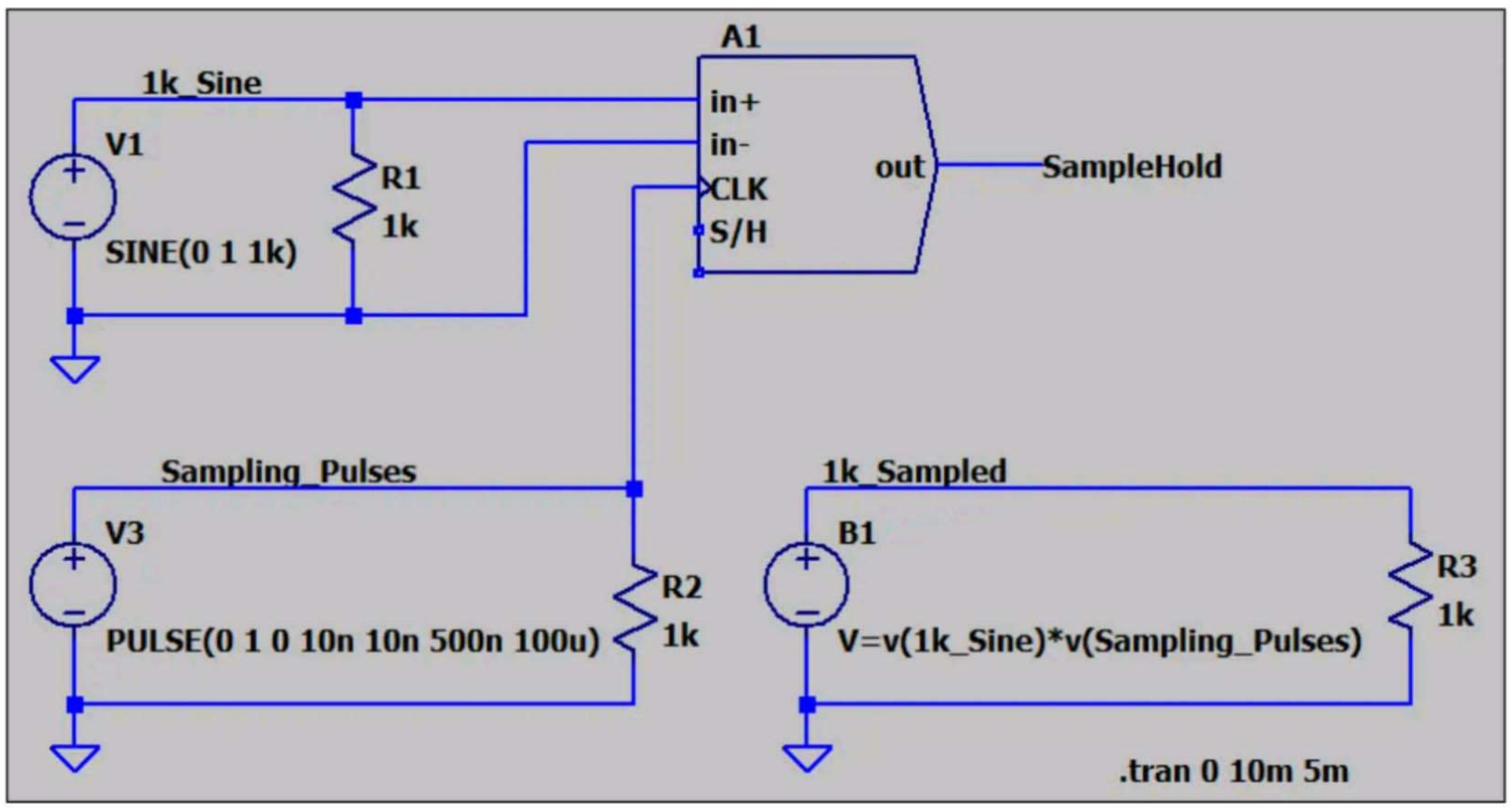


Fig.4: This LTspice schematic creates a sample pulse and sample-and-hold waveforms.

that produces a voltage or current proportional to the binary value.

There are various ways to do this, for example, a set of binary-weighted resistors or current sources, or ladders of resistors. When the register value is changed, the output will quickly step to the new value corresponding with the register contents. The output will be held constant between register changes, apart from the time it takes the circuit to react to register updates.

It is useful to be able to create a sample-and-hold waveform in LTspice, for example, to simulate the response of a reconstruction filter. In the July article, we simulated the sample-and-hold process using a couple of approaches.

The first was a simple switch and capacitor following the basic structure of a sample-and-hold circuit, and the second used a model of a real ADC, simulating its sampling process. In the latter case, the output is an analog representation of the converted voltage, so it produces a sample-and-hold type analog waveform.

The sample device

There is another option for a creating a sample-and-hold waveform in LTspice, which we will use here (see Fig.4). LTspice has a special function called "sample", which is a behavioural sample-and-hold model found in the Special Functions folder of the component tool.

This is an undocumented LTspice "A" type device (these have instance names starting with a letter A in the same way that, for example, resistor names all start with letter R).

There is general information on these devices under "A. Special Functions" in the LTspice help, but the sample device is not included.

However, some information is provided in an example schematic in the LTspice download, and the LTwiki page at ltwiki.org has information on this and other undocumented devices.

The sample device has two modes of operation that depend on which control input is used (only one may be used at a time). The `CLK` input samples the input voltage and transfers it to the output when a positive edge (logic 0 to 1 change) occurs – this is used here.

The `S/H` input provides track-and-hold functionality (like the capacitor-based circuit mentioned above). It operates in track mode at logic 1 and hold mode for logic 0.

The sample device has some parameters that can be set by right clicking the schematic symbol and entering the text in the form `parameter=value` into the Value box (with parameter/value pairs separated by spaces).

The parameters `vhigh` and `vlow` set the maximum and minimum output voltages, and default to 10V. The logic signals default to a threshold of 0.5 (1 above, 0 below). The logic threshold can be changed by modifying the `Ref` parameter.

The `Rout` parameter sets the output resistance and defaults to 1kΩ. This could cause problems with loading in some circuits if users are not aware of this. All the parameters are at their defaults for the circuit in Fig.4.

DAC waveforms

The circuit in Fig.4 creates a sample pulse train for a 1kHz sinewave sampled at 10kHz in a similar manner to previous examples.

The sampling pulses also clock the sample-and-hold device, which creates a sampled-and-held version of the input waveform, as would be seen at the output of a DAC. The results are shown in Fig.5.

The upper pane shows the original sinewave, while the second one below it is the sample pulse train and the third is the sample-and-hold (zero-order hold) waveform.

The bottom pane shows the sample pulses and zero-order hold waveform together. The hold value changes to a value equal to each sample pulse amplitude, and remains at that level until the next sample point.

We can use the circuit in Fig.4 to apply signals like a DAC output to a reconstruction filter to observe the performance of

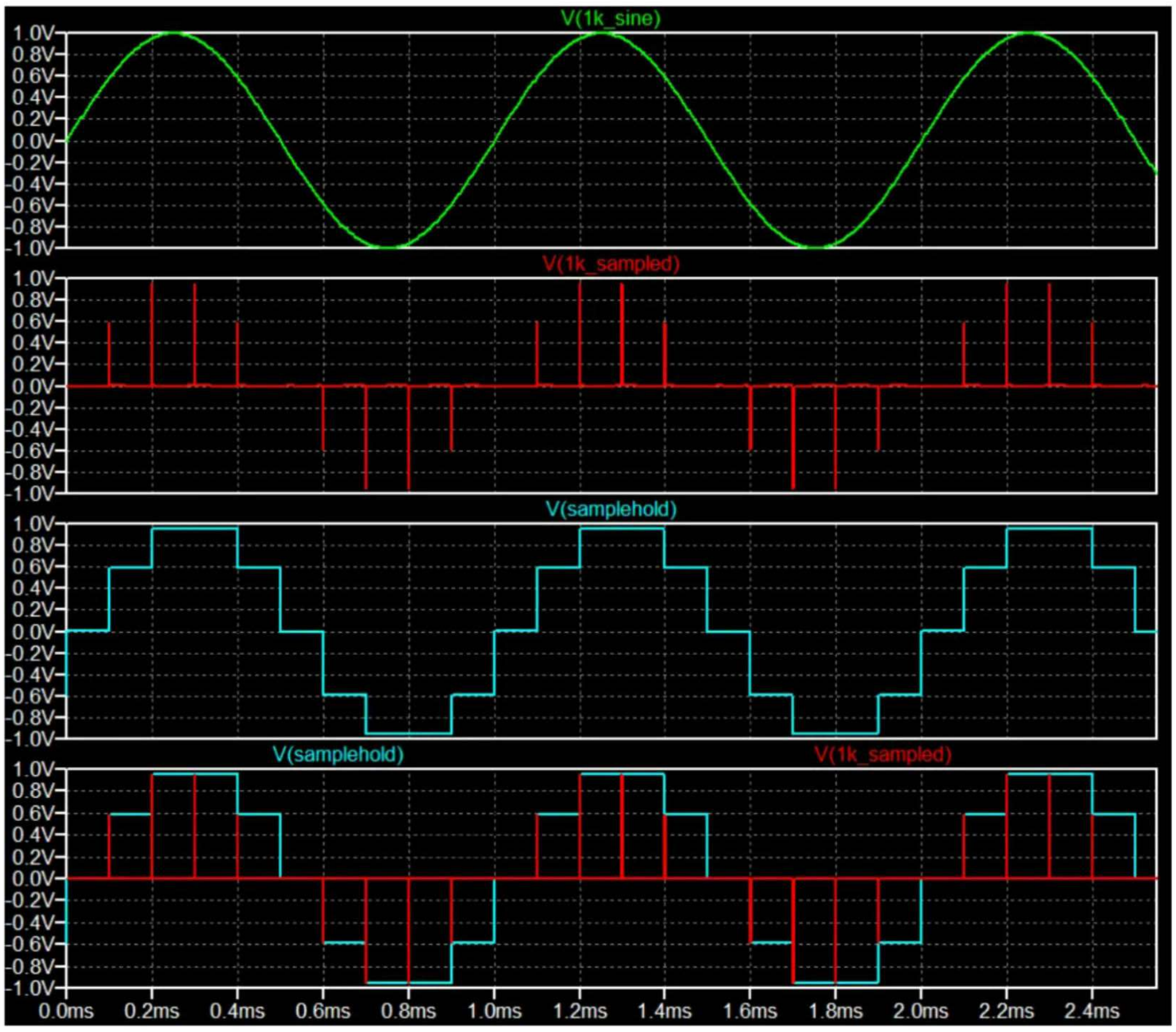


Fig.5: These plots show the output of LTspice from the simulation shown in Fig.4.

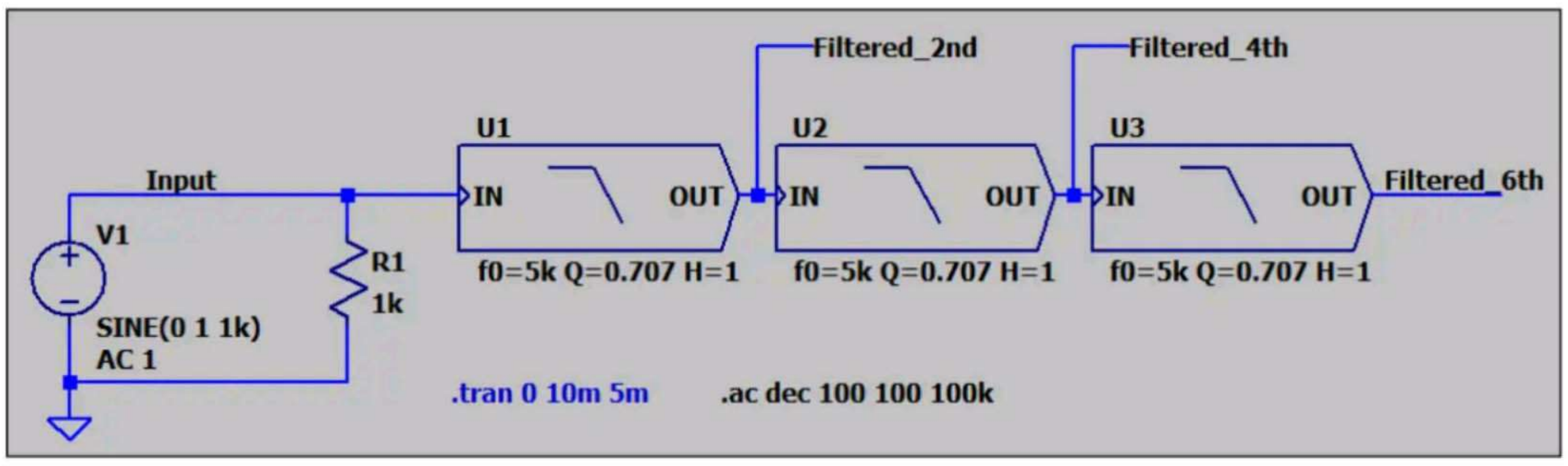

Filtered_2nd
Filtered_4th
U1
U2
U3
Input
IN
OUT
Filtered_6th
V1
R1
1k
SINE(0 1 1k)
AC 1
f0=5k Q=0.707 H=1
f0=5k Q=0.707 H=1
f0=5k Q=0.707 H=1
.tran 0 10m 5m
.ac dec 100 100 100k

Fig.6: An LTspice schematic of cascaded filters for reconstructing the DAC output.

the filter. Fig.6 shows the filter we will use for this illustrating this.

Again, we are using behavioural models from the special functions part of the library to create the filter quickly and easily. Of course, it can be replaced by any specific circuit design that might be of interest.

Several of these behavioural filters are available; the one used here is **`2ndOrdLowpass`**, which as the name suggests, is a second-order low-pass filter.

The **`2ndOrdLowpass`** device has three parameters that can be set by right clicking the symbol. These are the cutoff frequency (**`f0`**), the quality factor (**`Q`**), which sets the filter response type, and the gain (**`H`**).

In this example, the cutoff frequency is 5kHz (half the sampling frequency), the Q factor is 0.707 for a Butterworth-type response, and the gain is 1. Three of these filters are used in series to create second-, fourth- and sixth-order responses. Plots of the frequency responses from running an AC simulation on Fig.6 are shown in Fig.7.

Reconstruction filter simulation

Fig.8 shows the filters connected to the sample-and-hold output from Fig.4. The top pane shows the zero-order hold (DAC output) signal, while the lower pane shows the filter outputs with the same colours as Fig.7 (magenta for second order, yellow for fourth order and orange for sixth order).

Given that the sampled signal was an ideal sinewave, the output of the reconstruction filter should also be a perfect sinewave. The second-order filter clearly fails to do this; the waveform is distinctly distorted. The fourth-order filter is much better, but these is still some distortion just about visible. The sixth-order filter output looks very good in this image.

The three waveforms are offset in time due to the delays introduced by each filter stage.

Looking at the shape of sinewave does not provide much information about the quality of the signal unless it is quite heavily distorted. The spectrum of a signal, either from simulation or measurement using test equipment, provides a clearer picture of the signal quality and the nature of the any issues (eg, which unwanted frequencies are dominant).

Fig.10 shows the spectra of the filter outputs from Fig.9. To obtain good results, the simulation time was extended, the minimum timestep reduced, and the SPICE directive (**`.option plotwinsize=0`**) was added to prevent compression of the waveform data (shown as comments in Fig.8).

The spectra show the 1kHz wanted signal and the spectral images centred on one, two and three times the sample frequency. The 1kHz signal appears above and below the sample frequency multiples in the images, for example, at 9kHz and 11kHz for the first image centred on 10kHz. The three plots show the relative reduction amounts of the spectra images achieved by each filter.

In this example, the signal frequency is moderately close to the Nyquist rate and the filter cutoff frequency is set to be at the Nyquist frequency. We see that a reasonably high-performance filter (sixth-order) may be needed. In practice, it depends on the desired level of accuracy of the output signal, and we

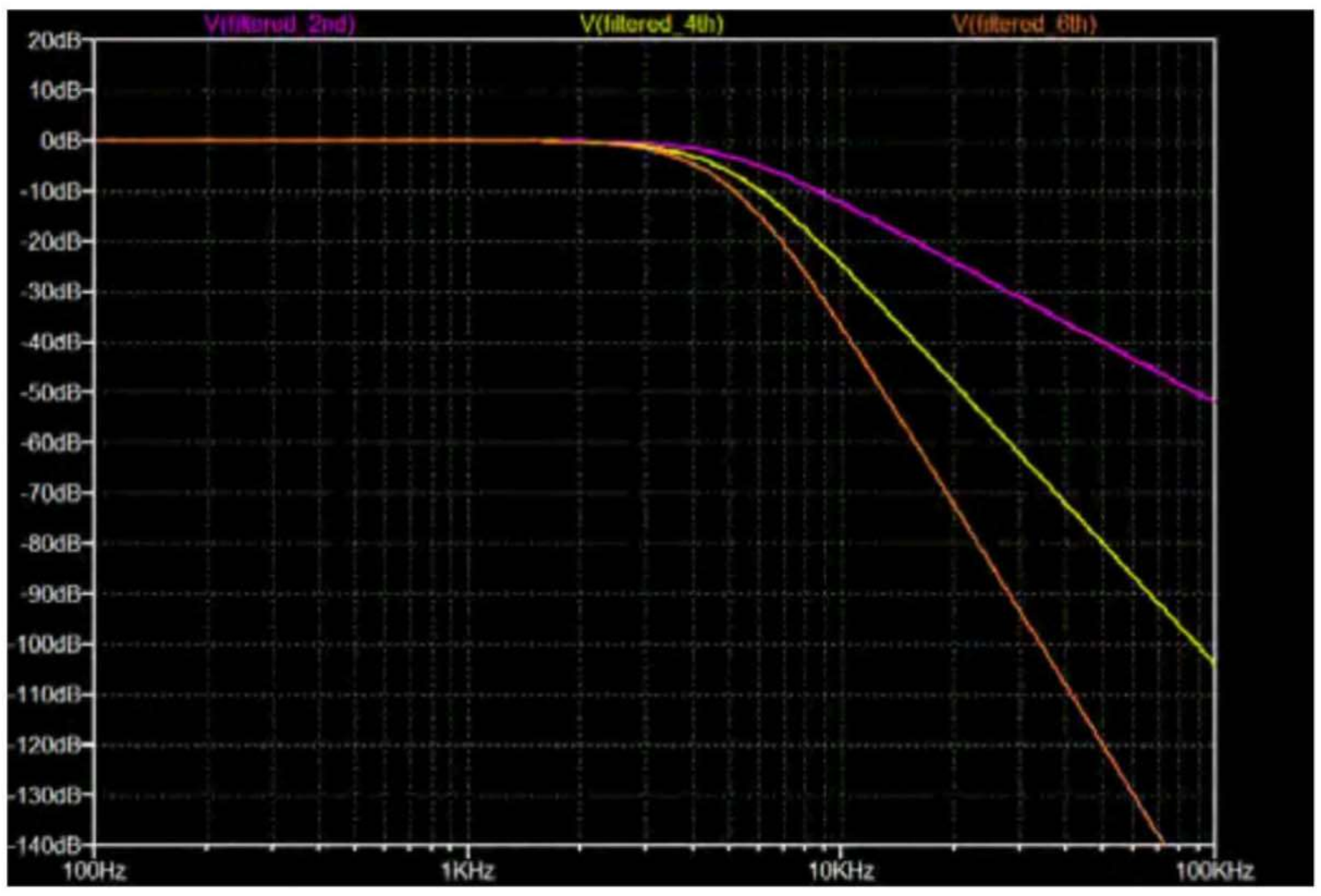


Fig.7: The filter responses from the circuit in Fig.6.

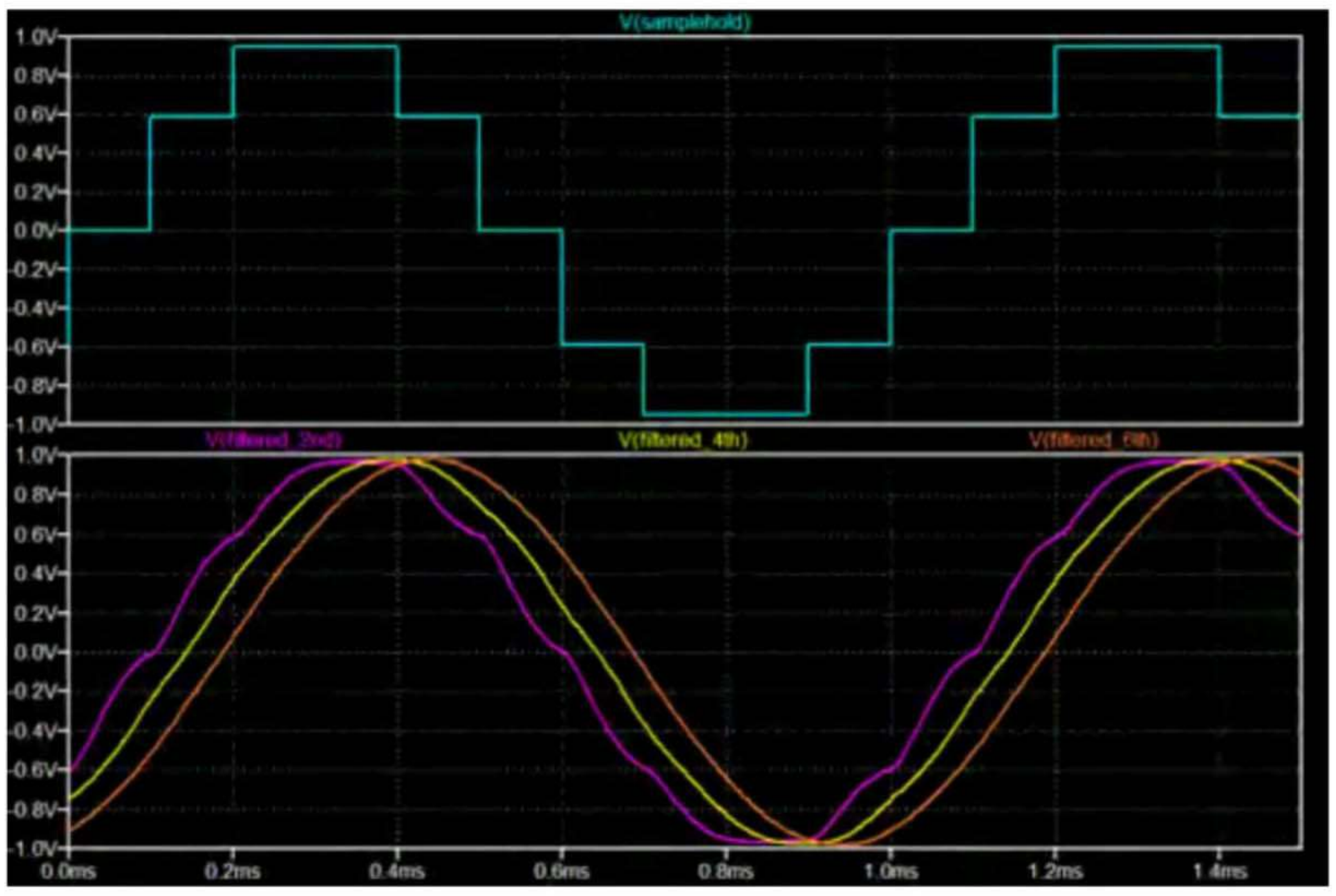


Fig.9: The simulation results from Fig.8. Note the varying accuracy of the reconstructed sinewaves.

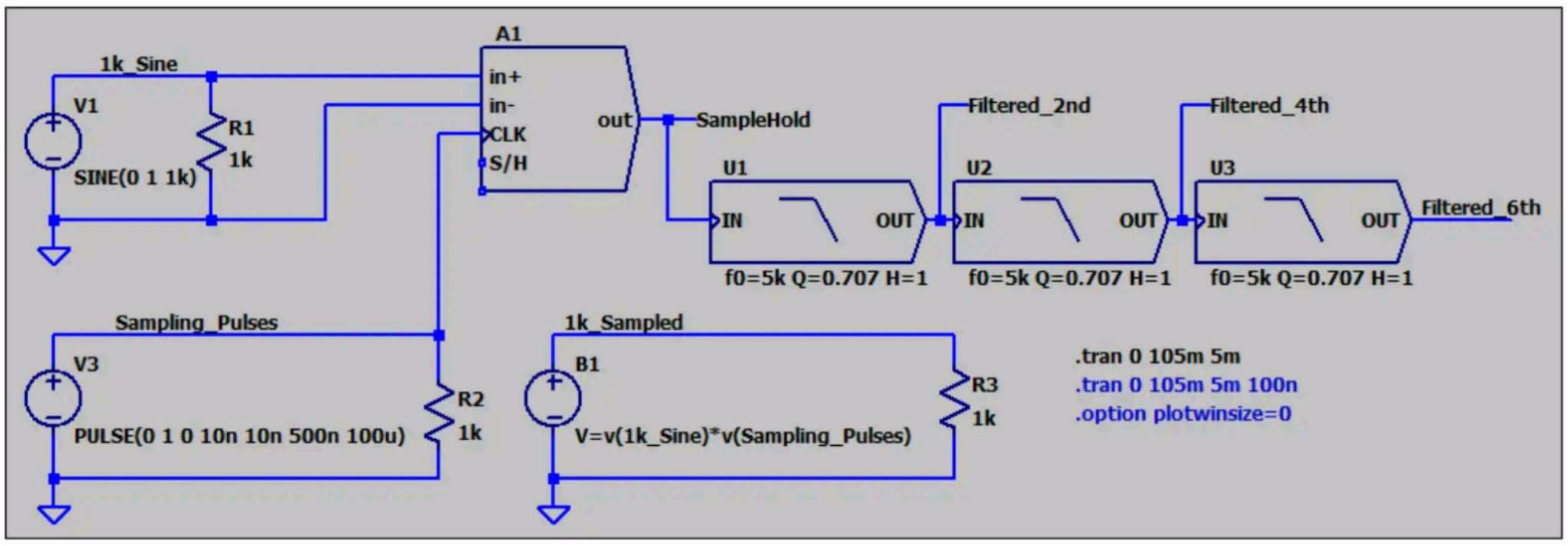

A1
1k_Sine
V1
R1
1k
SINE(0 1 1k)
in+
in-
CLK
S/H
out
SampleHold
Filtered_2nd
Filtered_4th
U1
U2
U3
IN
OUT
Filtered_6th
f0=5k Q=0.707 H=1
f0=5k Q=0.707 H=1
f0=5k Q=0.707 H=1
Sampling_Pulses
V3
PULSE(0 1 0 10n 10n 500n 100u)
R2
1k
1k_Sampled
B1
V=v(1k_Sine)*v(Sampling_Pulses)
R3
1k
.tran 0 105m 5m
.tran 0 105m 5m 100n
.option plotwinsize=0

Fig.8: This LTspice circuit illustrates the filtering of a zero-order hold signal.

would have the option of using filter types with faster cutoffs than the Butterworth, that might be suitable at a lower order.

As mentioned above, we can reduce the filtering requirements by using a higher sampling rate. This is illustrated in Fig.11, which uses the same simulation as Fig.8 but with the sampling rate increased to 50kHz – this is achieved by changing the `Tperiod[s]:` parameter of the V3 pulse source from 100µs to 20µs. We see that the second- and fourth-order filters produce good sinewave outputs with the higher sampling rate.

This is the reason that most high-performance audio DACs these days are oversampling types, that operate at a multiple of the actual sampling rate to make the filter's job much easier. But that is a topic for another column.

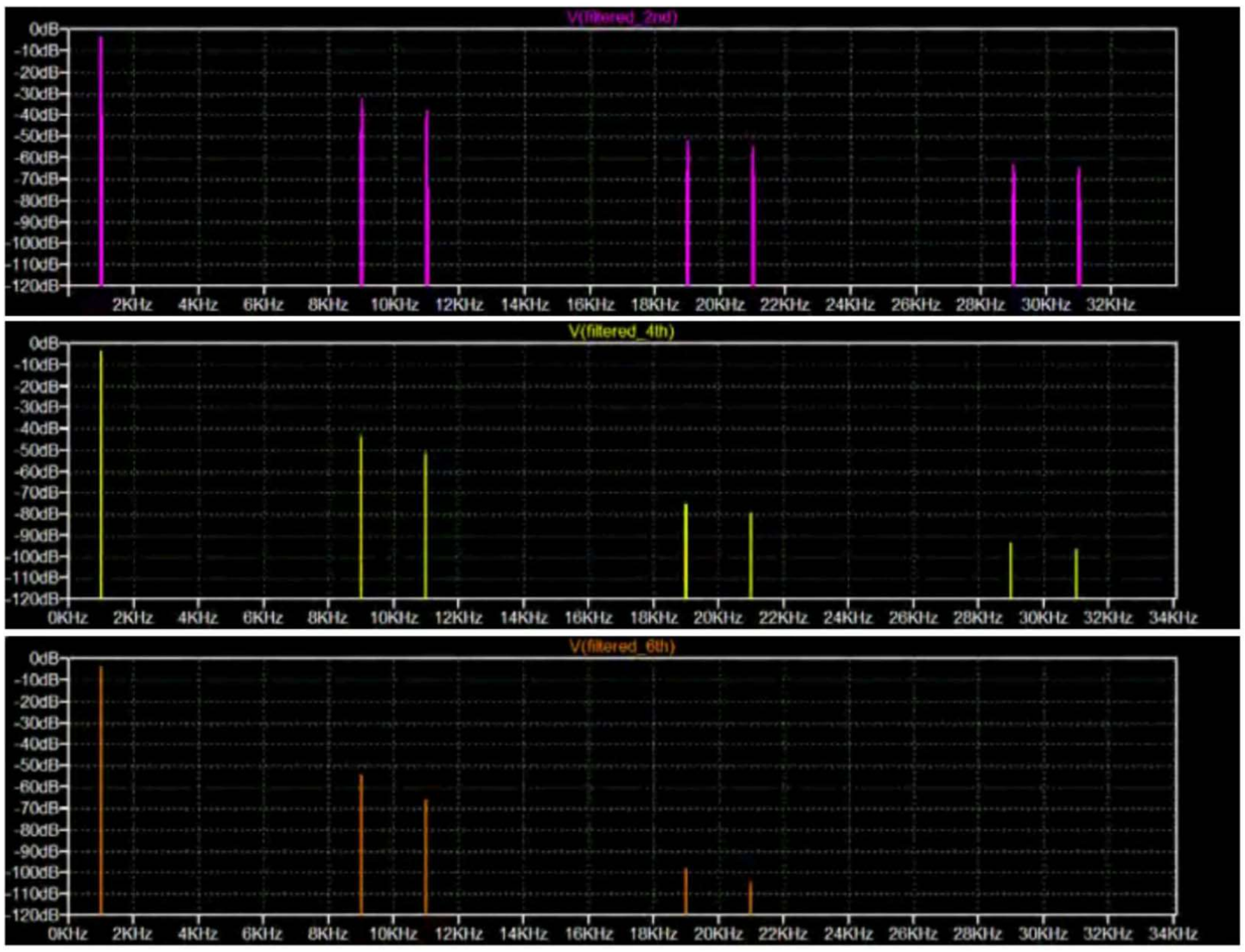


Fig.10: Spectral plots of the output signals from the filters shown in Fig.8.

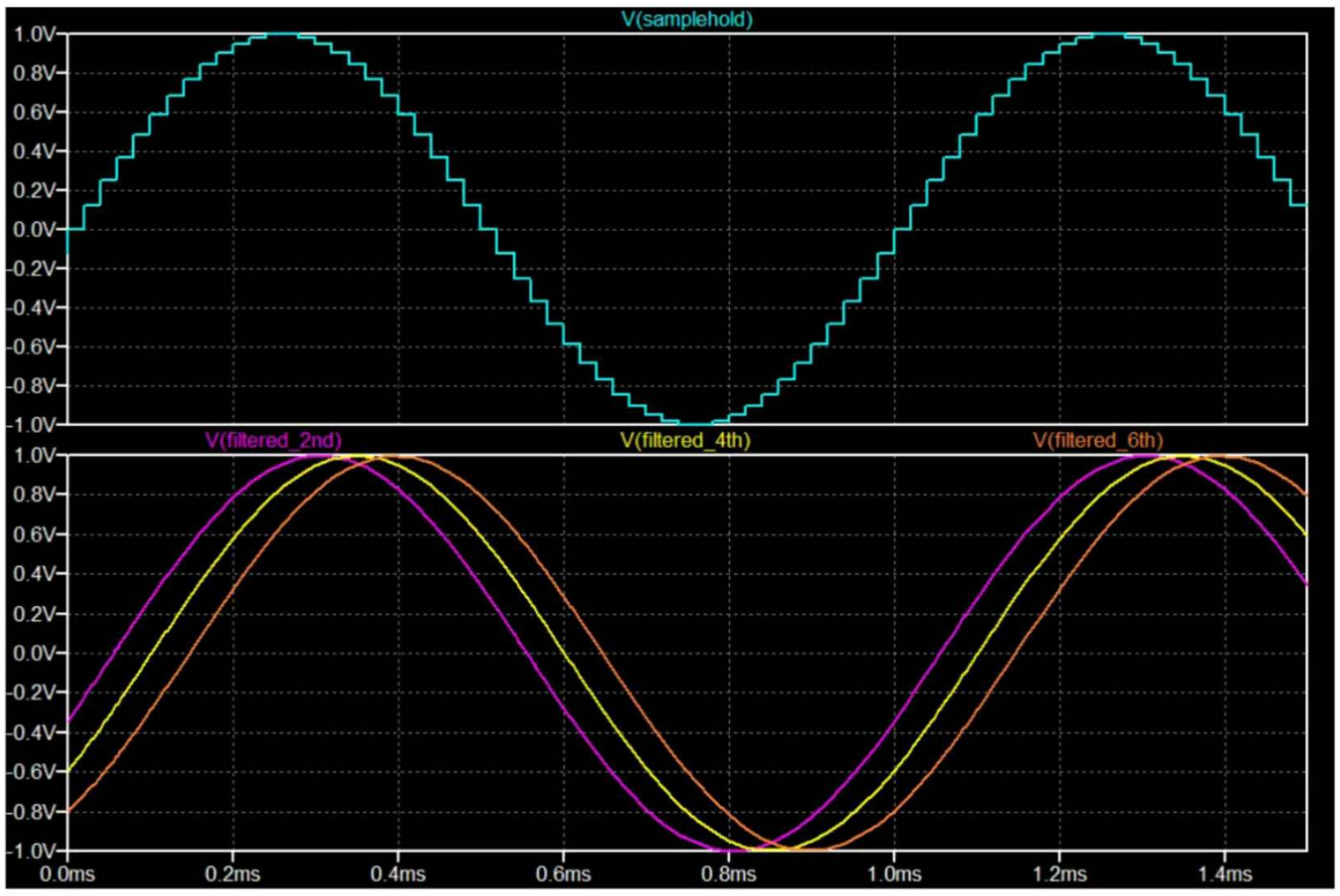


Fig.11: The simulation results from Fig.8 with the sampling rate increased to 50kHz.

DAC frequency response

We have shown some simulations that illustrate the shape of the DAC output waveform and the effects of filter response and sampling rate on the reconstruction of a sinewave.

This seems to indicate that if we have a sufficient sampling rate, a filter with a flat passband and a sufficiently sharp cutoff, we can more-or-less perfectly reconstruct the sampled signal, particularly if we can get close to the ideal case shown in Fig.2. However, this is the not whole story of using a DAC to reproduce a sampled waveform.

A waveform's spectrum depends on the shape of the waveform, but the shape of the idealised sample impulses and zero-order hold output from the DAC are not the same. This implies the DAC does not have a flat frequency response. It will modify the spectrum of the waveform away from that of the sampled data from the digital processing part of the system.

Thus, the DAC can distort the waveform even in a system in which the samples are passed to an ideal DAC and brick-wall reconstruction filter. This effect is worse for lower sampling rates relative to the original signal, so increasing the sampling rate will mitigate the effects of the DAC's frequency response. It also possible to compensate for the DAC's response by suitable additional filtering.

The sinc function

The shape of a DAC's frequency response relates to a mathematical function denoted sinc (pronounced in the same way as "sink"). The sinc notation is attributed to British mathematician Philip Woodward in 1952 (the function itself was used well before this). It is said to stand for "cardinal sine" (or *sinus cardinalis* in Latin), but that name was not used by Woodward and there is debate over the origin of the name – which might appeal to those who like exploring maths history and etymology (see https://pemag.au/link/ac00).

The sinc function is defined as:

$$\mathrm{sinc}(x) = \frac{\sin(x)}{x}$$

When x is zero, we get 0/0, which is mathematically problematic (due to the division by zero), so sinc(0) is defined to be 1. This is because sinc(x) tends towards 1 as x tends towards 0. This mathematical trick is called a removable singularity. A variant is the normalised sinc function, which is defined as:

$$\mathrm{sinc}_\pi(x) = \frac{\sin(\pi x)}{\pi x}$$

...again with $\mathrm{sinc}_\pi(0)$ defined to be 1.

This function occurs frequently in DSP and telecommunications. It has the property that the points at which $\mathrm{sinc}_\pi(x)$ is equal to zero occur at integer values of x and (for those who understand calculus), it has a value of 1 when integrated from minus to plus infinity (hence the term normalised).

An obvious question is: why is a DAC's frequency response a sinc function? As we have mentioned previously, the mathematical process of obtaining a spectrum from a waveform is called the Fourier transform.

The answer, without going into too much detail, is that the Fourier transform of a rectangular function is a sinc function and the DAC output (zero-order hold) is effectively a set of rectangular pulses in the time domain, so the process

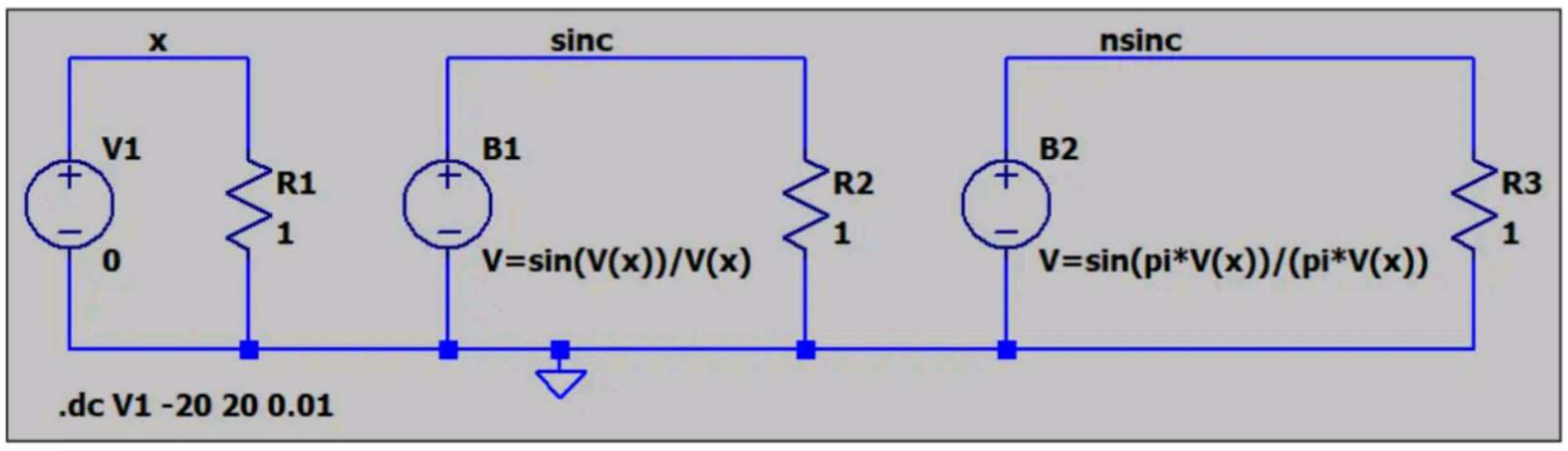


Fig.12: An LTspice schematic for plotting the sinc() function.

Simulation files

Most months, LTSpice is used to support descriptions and analysis in Circuit Surgery.
The examples and files are available for download from the PE website:
https://bit.ly/pe-downloads

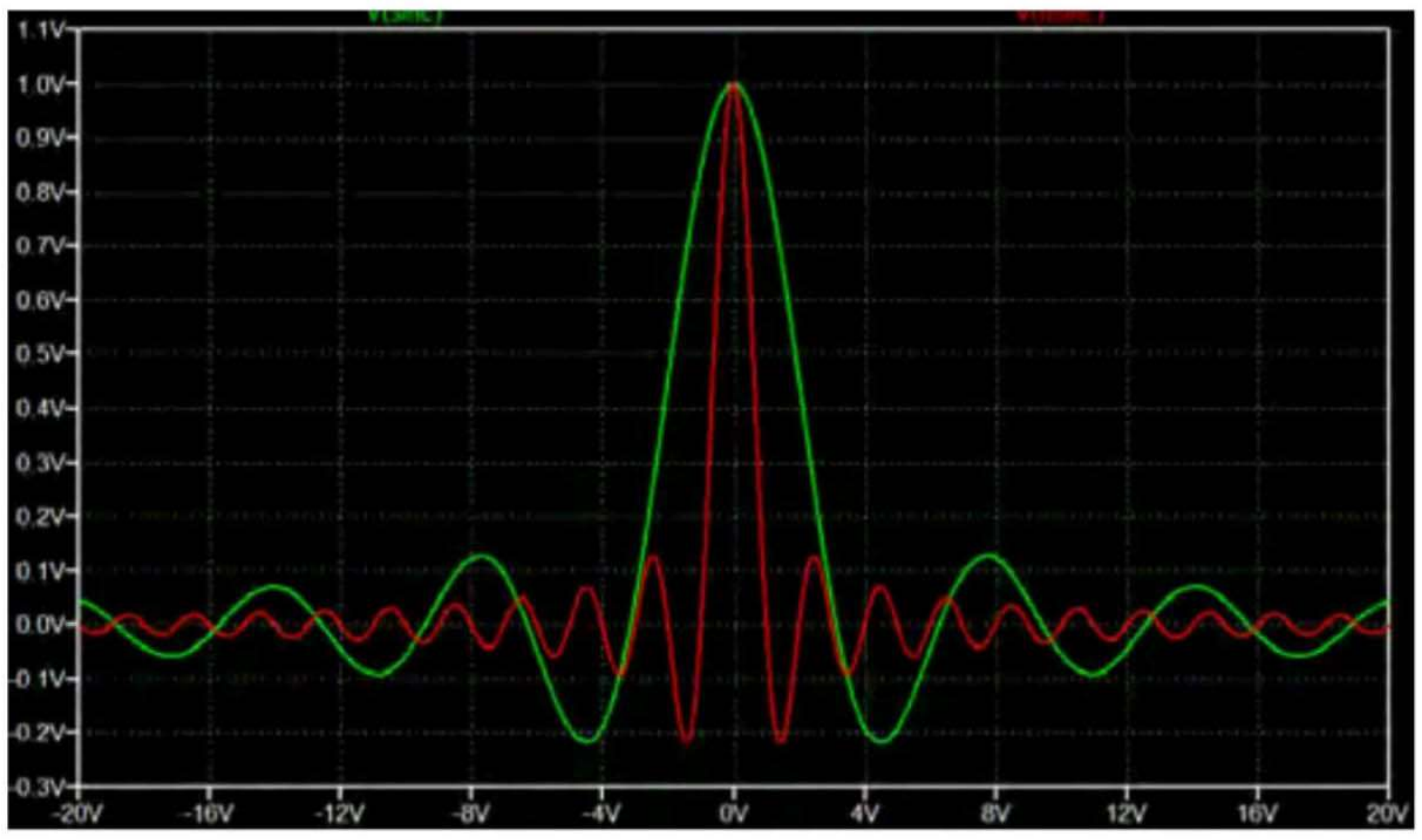


Fig.13: Sinc (green) and normalised sinc (red) plotted using the circuit shown in Fig.12. The values are numbers, not voltages.

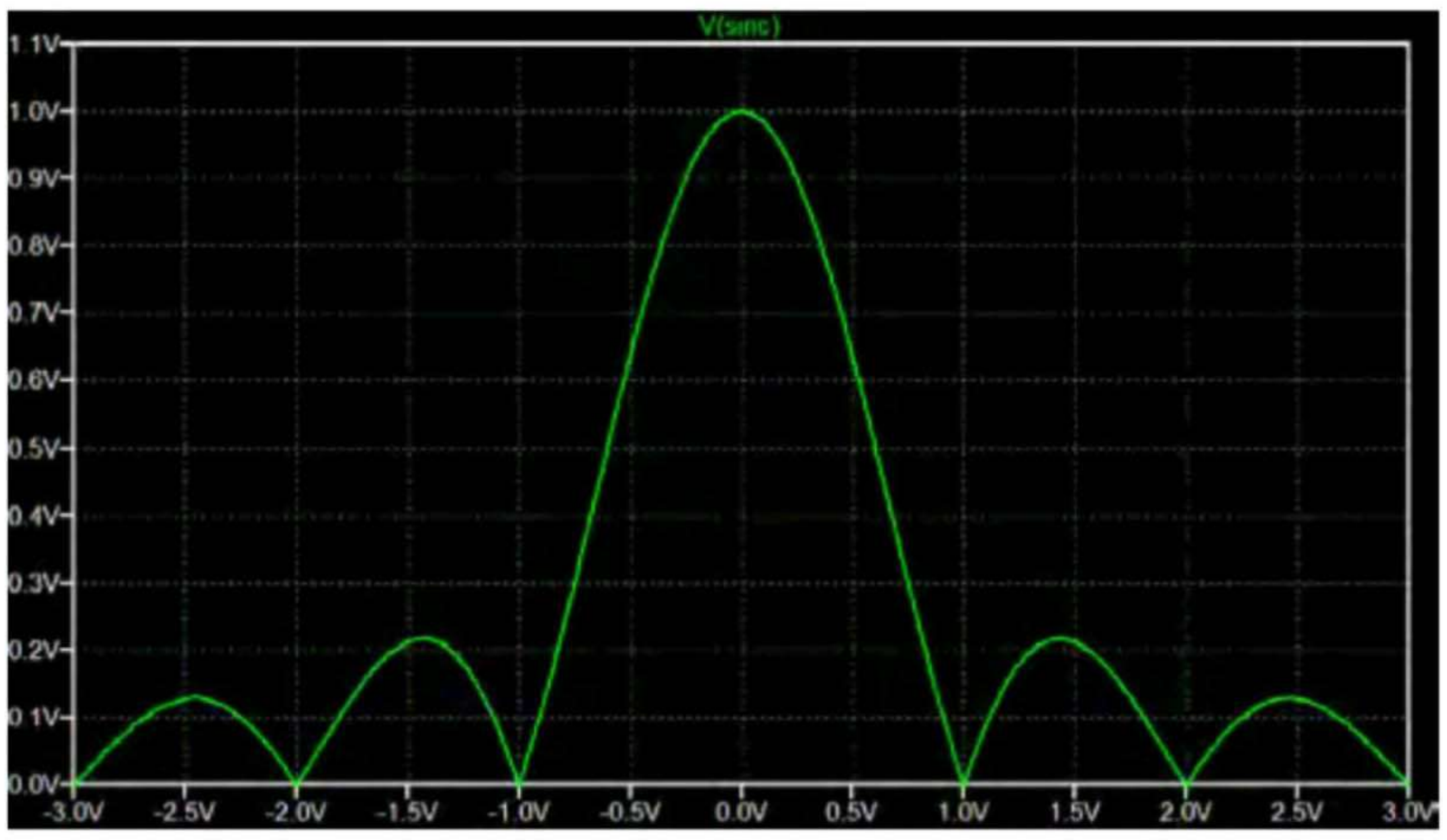


Fig.14: The DAC frequency response plotted by LTspice. The values are numbers, not voltages; the X-axis is f/f_s.

of outputting the samples with a DAC in the time domain translates to a sinc function in the frequency domain (the frequency response).

As well as the frequency response of a DAC, the sinc function is important in the implementation of digital filters, so we will come across it again later in this series.

Plotting the sinc function

The shape of the sinc function can be observed by plotting it in LTspice. This is not really a normal use for LTspice, but it is straightforward using behavioural sources (see Fig.12) and may be useful if you do not have access to or familiarity with other tools.

The plots are achieved by using a voltage source, V1, to produce a signal **x** (for the x-axis) controlled by a DC sweep analysis, which steps the value of **x** over the desired range (-20 to 20 in this case). The behavioural sources calculate the value of the sinc function at each point in the DC sweep.

The expression used in the behavioural sources for the sinc and normalised sinc function respectively are:

```
V=sin(V(x))/V(x)
V=sin(pi*V(x))/(pi*V(x))
```

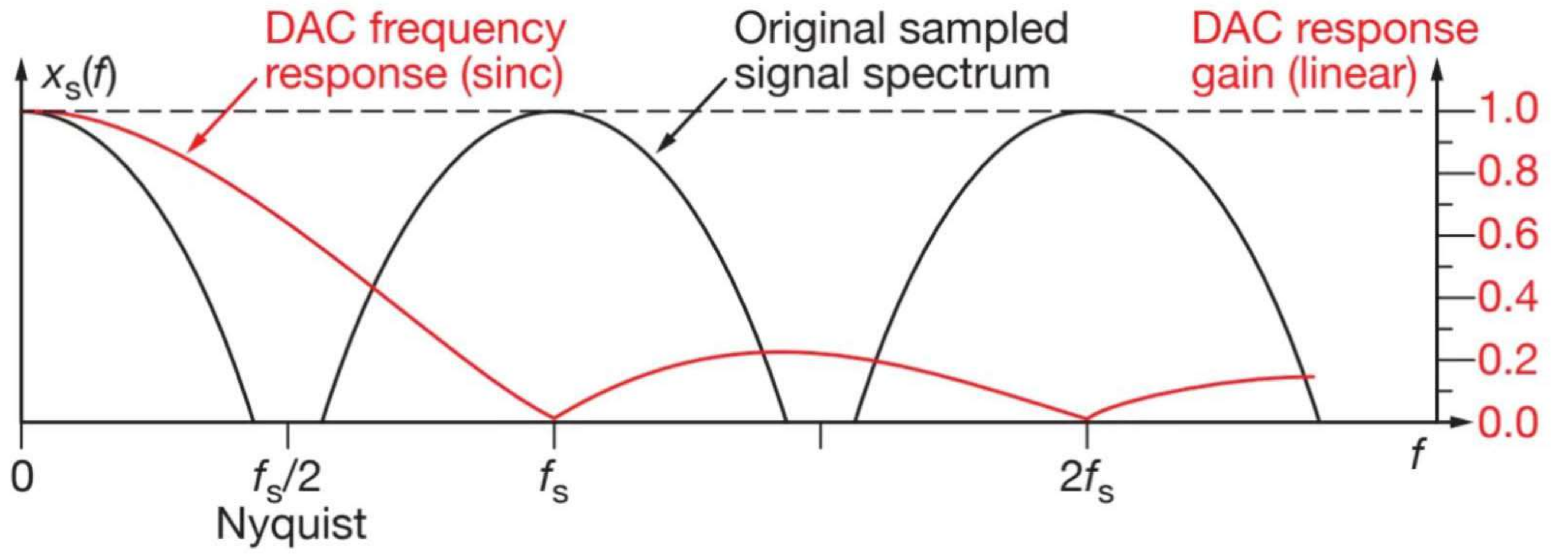


Fig.15: The DAC frequency response (sinc function) with a generic sampled signal spectrum, using a linear scale for the DAC response.

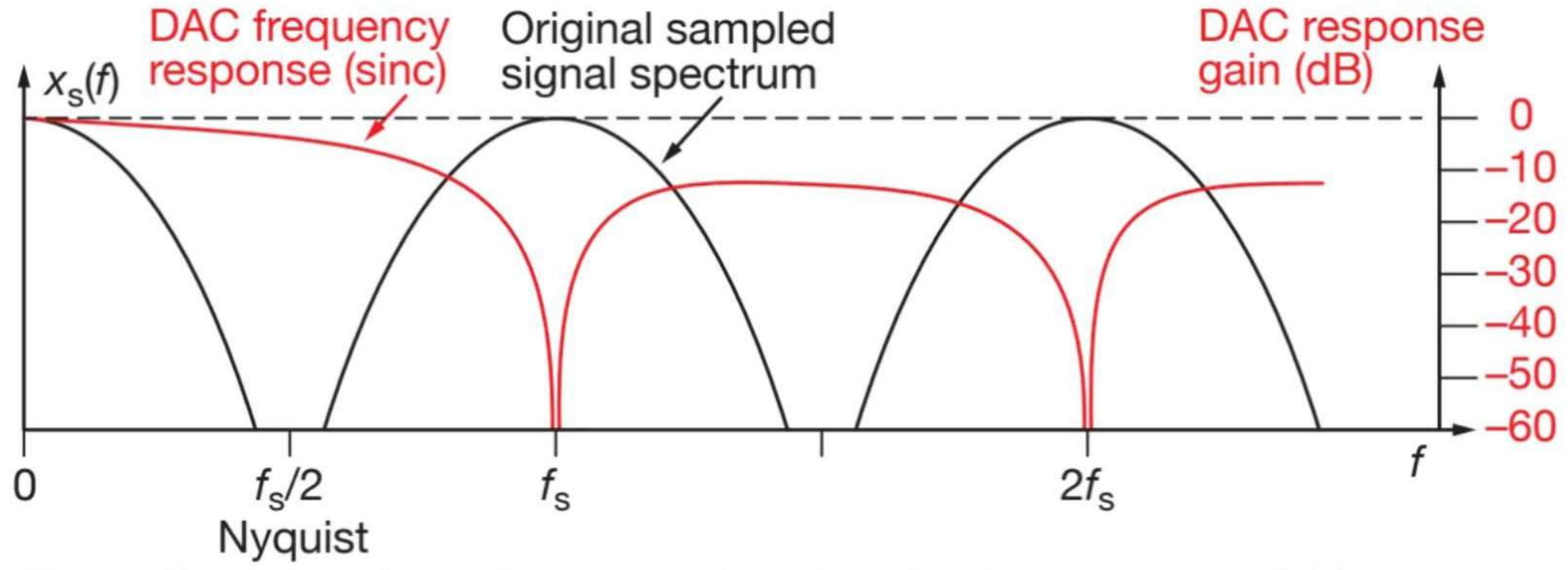


Fig.16: The same plot as Fig.15 but with a logarithmic scale for the DAC response.

The results are shown in Fig.13 – these are just numerical values, although LTspice plots them as voltages.

The actual DAC frequency response is obtained using $x = f/f_s$ in the normalised sinc function and taking the absolute value (ignoring the sign). f is the frequency in the DAC output signal spectrum and f_s is the sample frequency. This is plotted in Fig.14 using the circuit in Fig.12 modified with the following expression in the B1 behavioural source:

```
V=abs(sin(pi*V(x))/(pi*V(x)))
```

The x axis represents f/f_s and we see that the value is zero at integer multiples of the sample frequency ($f/f_s = 1, 2, 3$ etc).

Fig.15 shows the DAC frequency response superimposed on the generic sampled signal spectrum used in Figs.2 & 3. Only the positive part is shown to save space. The plot shows how the DAC response aligns with the signal spectrum.

Remember that we only want the part of the spectrum up to the Nyquist frequency ($f_s/2$); the rest should be removed by the reconstruction filter. Over the wanted range, we see that the effect of the DAC's frequency response is to attenuate the higher signal frequencies.

Fig.15 plots the DAC response on a linear scale; it is taken directly from the relevant part of Fig.14, scaled appropriately. Fig.16 shows the DAC response plotted on a decibel (dB) scale. The DAC response goes to zero at f_s and $2f_s$, which is minus infinity on the dB scale and beyond the axis range of –60dB, hence the gaps.

We will continue to look at DACs and their output signals next month. PE

Win a Microchip dsPIC33A Curiosity Platform Development Board

Practical Electronics is offering its readers the chance to win a Microchip dsPIC33A Curiosity Platform Development Board (EV74H48A). Even if you don't win, receive a 15%-off voucher, plus free shipping for one of these products.

The dsPIC33A Curiosity Platform Development Board (EV74H48A) is a full-featured development and demonstration platform that enables customers to explore the capabilities of the dsPIC33A Digital Signal Controller (DSC) family. Its 32-bit architecture, double-precision floating-point unit and DSP engine enable fast computation in time-critical applications.

Built around a 32-bit CPU operating at 200MHz, the dsPIC33A family's advanced core includes a Double-Precision Floating-Point Unit (DP FPU) and DSP instructions for numerically intensive tasks in closed-loop control algorithms. The architecture offers high performance and precise real-time control with a comprehensive development tool ecosystem to streamline the design process.

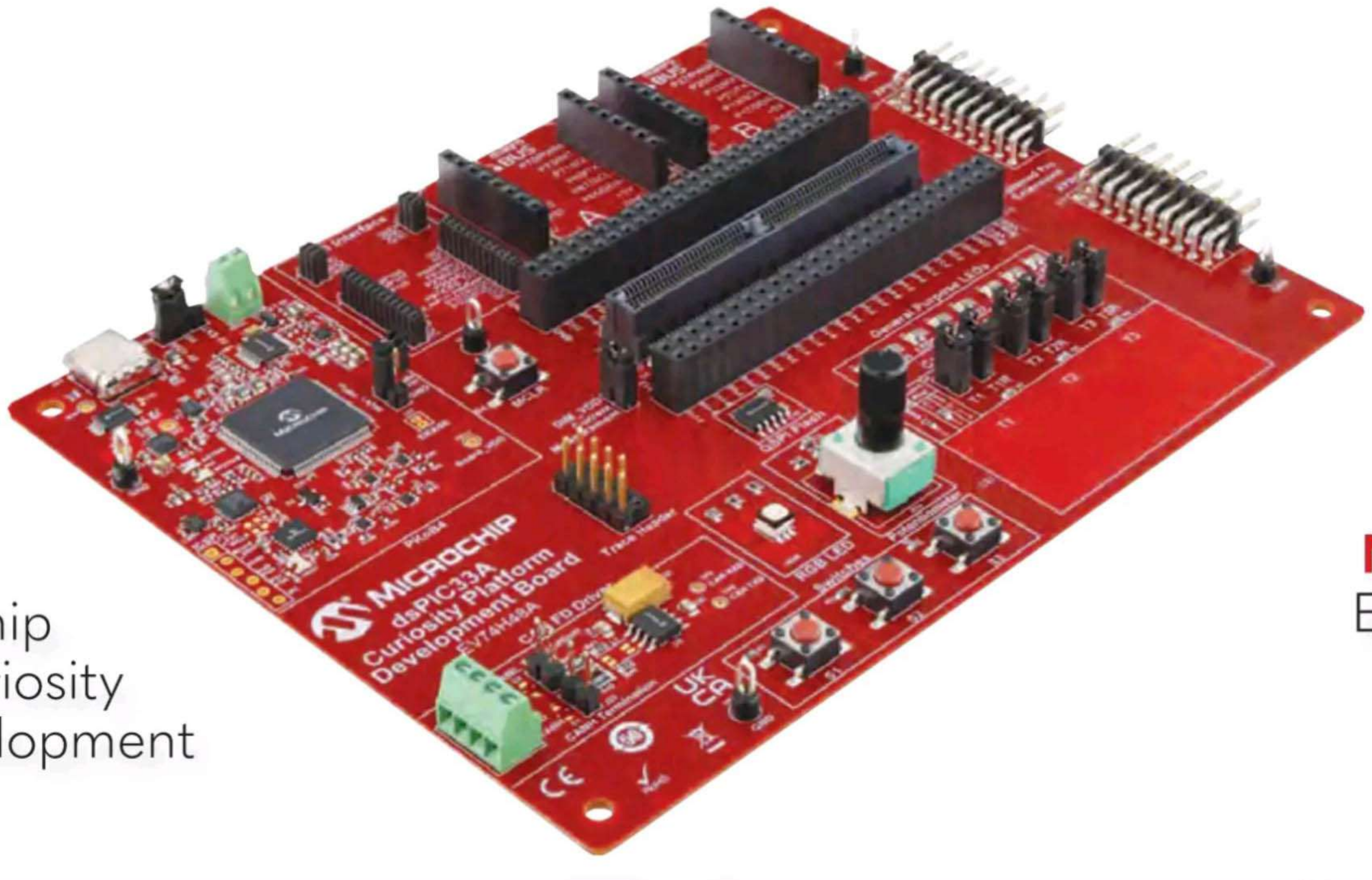
Worth $98 (approx £75) each

The dsPIC33A family of DSCs with low-latency responsiveness is designed for high-performance and real-time precision control applications. These DSCs combine the advantages of MCUs with the signal processing capabilities of Digital Signal Processors (DSPs), making them suitable for a wide variety of applications that require quick control loop computations using multiple sensor inputs.

The board has mikroBUS™ and XPLAINED PRO interfaces, allowing companion development boards to be added, such as QT Touch extension boards, BIST XPRO extension kits, various sensors and interfaces via Click Boards™. A QSPI connection to a 2Mbit SQI Flash EEPROM supports external non-volatile data storage.

It comes with one RGB LED, a 10kΩ potentiometer, three push buttons, eight LED's and three capacitive touch pads. Each of these can be programmed for any purpose to create the desired interaction between the user and the board behaviour.

Free-to-enter competition



Microchip
Win a Microchip dsPIC33A Curiosity Platform Development Board

Microchip
EV74H48A

How to enter

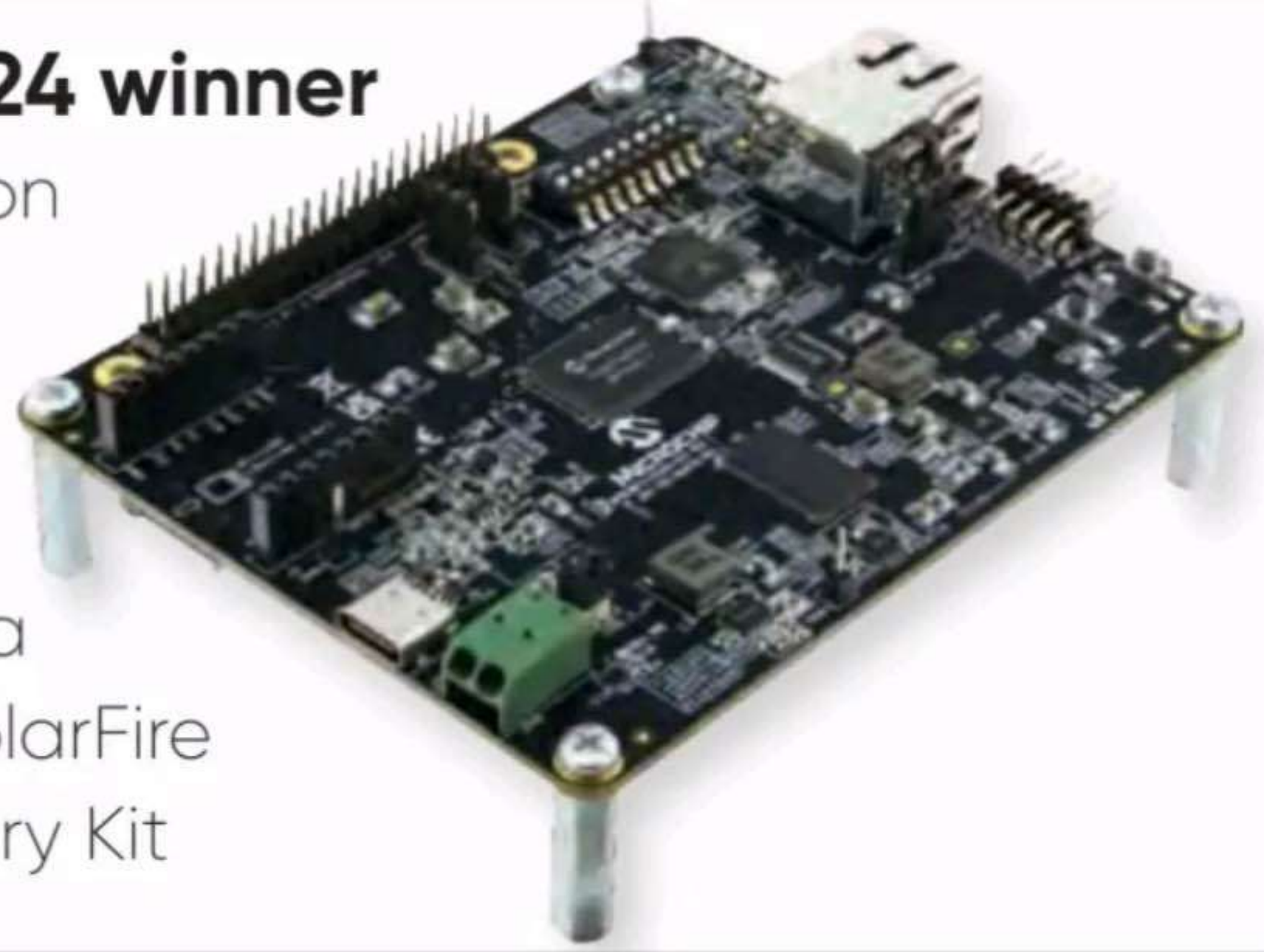
For your chance to win a Microchip dsPIC33A Curiosity Platform Development Board or receive a 15%-off voucher, including free shipping, enter your details in the online entry form at: https://page.microchip.com/PE-33A.html

Closing date

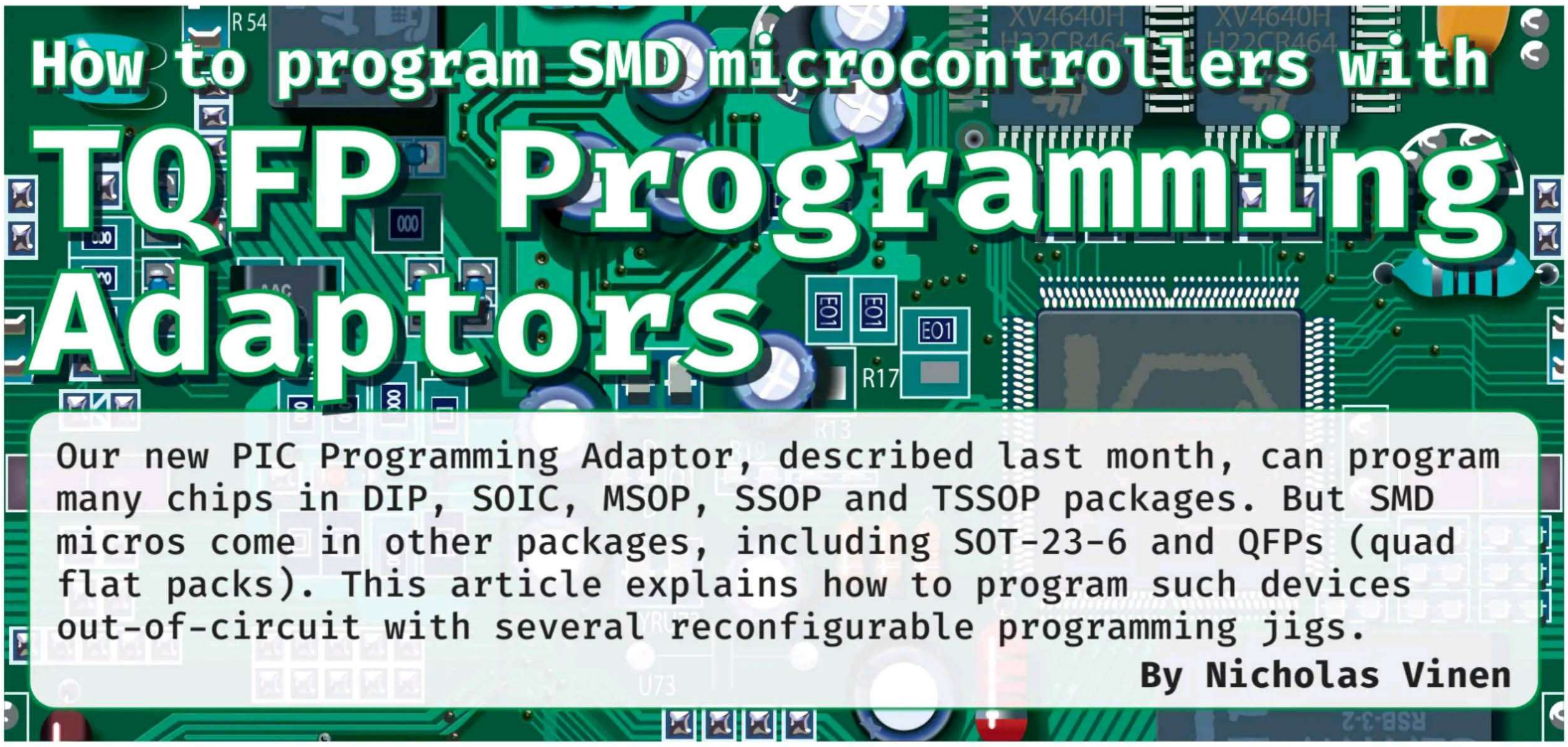
The closing date for this offer is 31 September 2024

August 2024 winner

James Walton



James won a Microchip PolarFire SoC Discovery Kit



How to program SMD microcontrollers with

TQFP Programming Adaptors

Our new PIC Programming Adaptor, described last month, can program many chips in DIP, SOIC, MSOP, SSOP and TSSOP packages. But SMD micros come in other packages, including SOT-23-6 and QFPs (quad flat packs). This article explains how to program such devices out-of-circuit with several reconfigurable programming jigs.

By Nicholas Vinen

These jigs are inexpensive and straightforward to make. Still, they are invaluable if you need to program SOT-23-6 or QFP microcontrollers before being soldered to a board, such as when the board lacks a programming header. They aren't limited to PICs; they will work with most microcontrollers in these packages, including AVRs and many ARM-based types.

We use jigs like these all the time to program chips we sell, including the ones below:

- The 64-pin PIC32MX470F-512H-120/PT programmed for the Micromite Plus (Explore 64).
- The 100-pin PIC32MX470F-512L-120/PF, also for the Micromite Plus (Explore 100).
- The 44-pin PIC16F18877-E/PT for our recent Wideband Fuel Mixture Display (WFMD).
- The 32-pin ATSAML10E16A-AUT for the High-Current Battery Balancer.

Those are all QFP chips but with different numbers of pins, so a separate jig is needed for each one. Note that sometimes QFP chips with the same number of pins can be different sizes, so you may need more than one jig with the same pin count. However, as our jigs are reconfigurable, you only need one of each, even if you need to program different chip types in that package.

Most QFP micros actually come in either TQFP ("T" stands for "thin") or LQFP ("low-profile") packages. The sockets we're specifying suit TQFP, although other types may be available.

Let's go through the jigs individually, from the fewest pins to the most.

SOT-23-6

This suits tiny PICs like the PIC10(L)F202-I/OT or PIC10(L)F322-I/OT that we used in our Remote Control Range Extender (January 2023).

These often need to be programmed out-of-circuit because they're typically used on very small PCBs that probably don't have much room for a programming header.

This is the simplest jig as it is just made of a commercial SMD to DIP adaptor (AliExpress pemag.au/link/abmu) plus five female-to-male jumper leads. It is shown in Photo 1, and the wiring is shown in Fig.1.

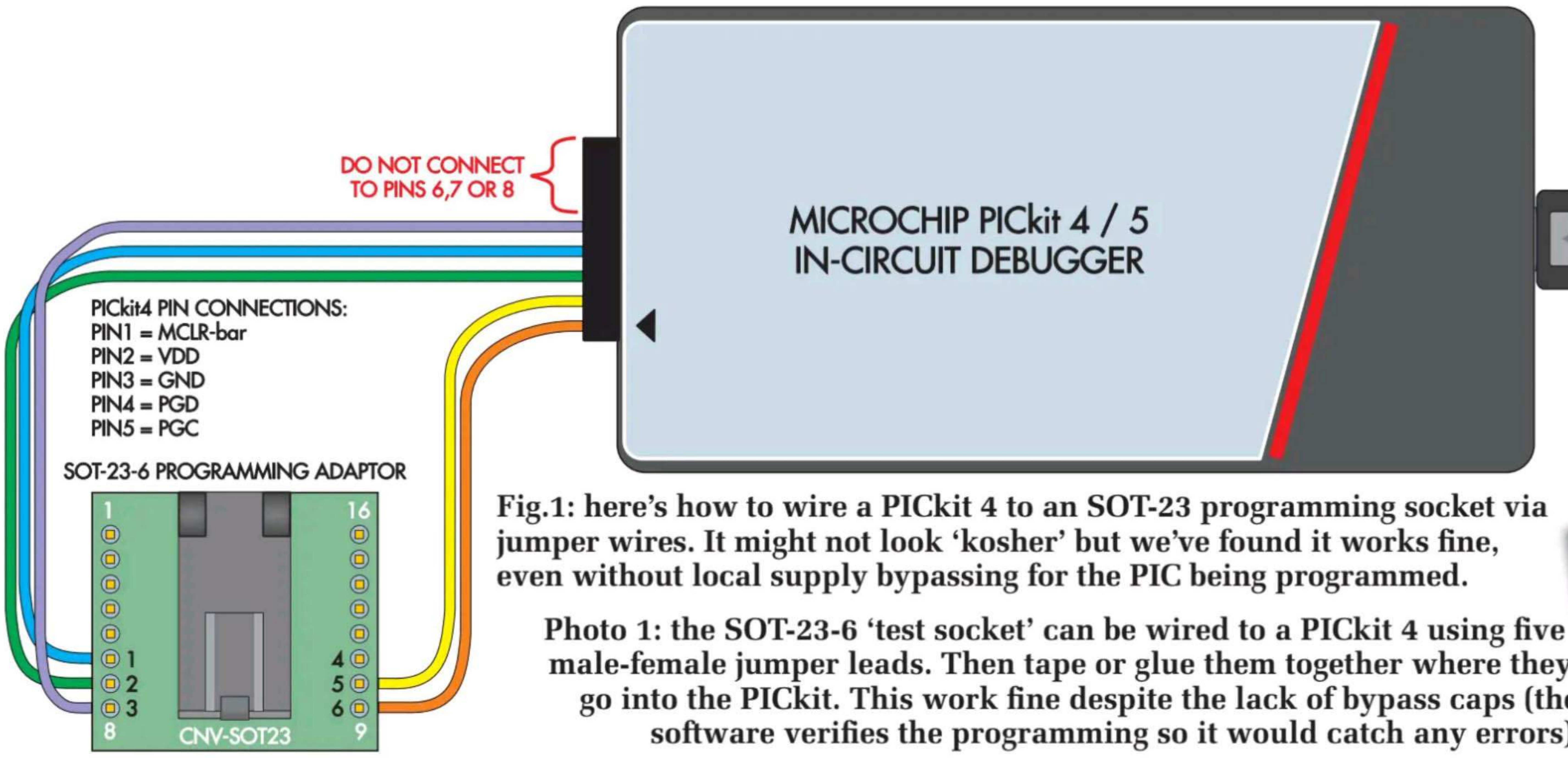


Fig.1: here's how to wire a PICkit 4 to an SOT-23 programming socket via jumper wires. It might not look 'kosher' but we've found it works fine, even without local supply bypassing for the PIC being programmed.

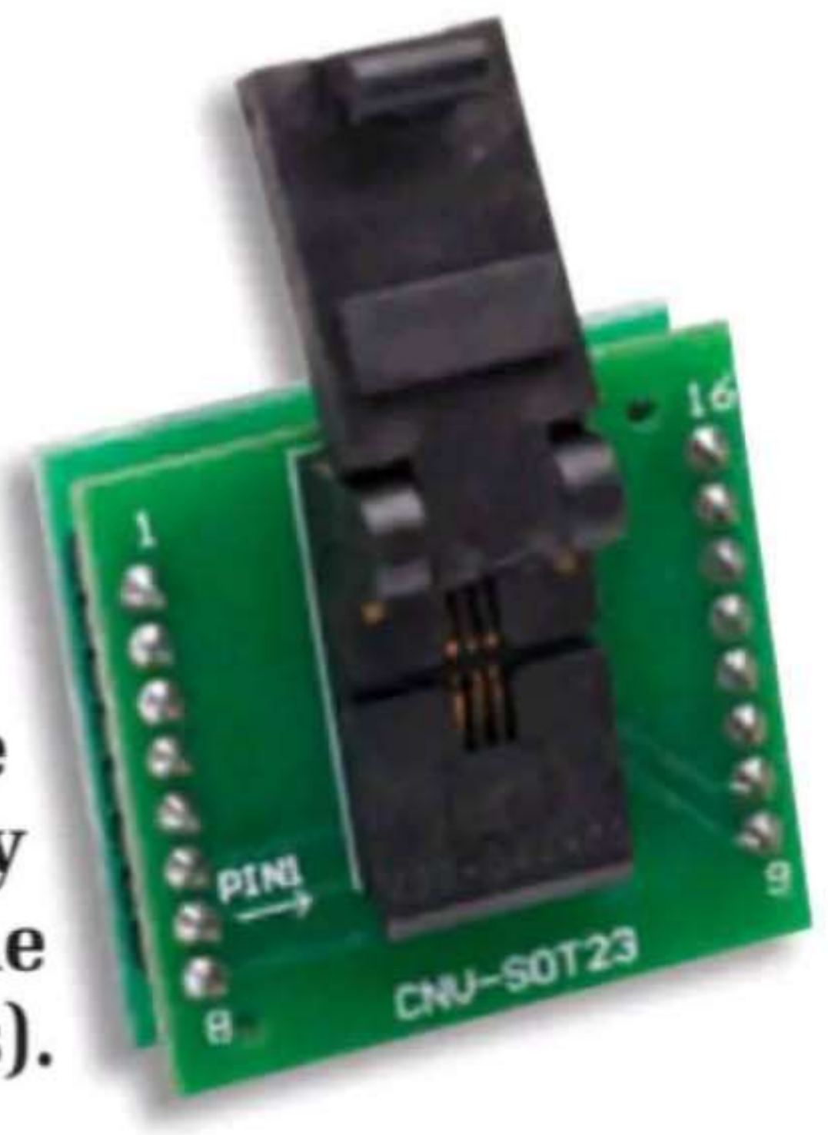
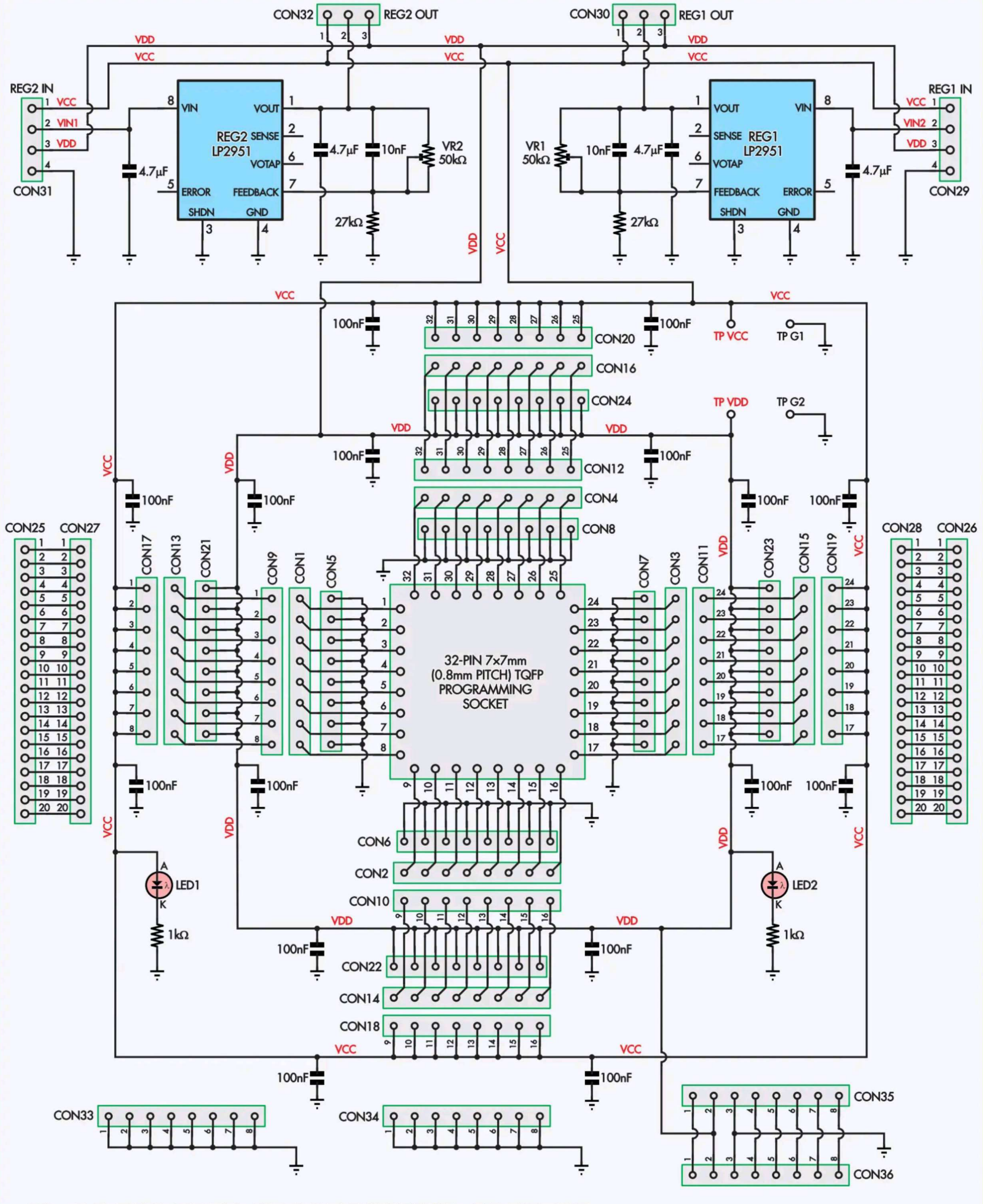


Photo 1: the SOT-23-6 'test socket' can be wired to a PICkit 4 using five male-female jumper leads. Then tape or glue them together where they go into the PICkit. This work fine despite the lack of bypass caps (the software verifies the programming so it would catch any errors).


CON32
REG2 OUT
CON30
REG1 OUT
VDD
VCC
REG2 IN
VIN1
CON31
REG1 IN
VIN2
CON29
REG2
LP2951
REG1
LP2951
VIN
VOUT
SENSE
VOTAP
ERROR
FEEDBACK
SHDN
GND
4.7µF
10nF
VR2
50kΩ
VR1
50kΩ
27kΩ
100nF
TP VCC
TP G1
TP VDD
TP G2
CON20
CON16
CON24
CON12
CON4
CON8
CON25
CON27
CON28
CON26
CON17
CON13
CON21
CON9
CON1
CON5
CON7
CON3
CON11
CON23
CON15
CON19
32-PIN 7x7mm
(0.8mm PITCH) TQFP
PROGRAMMING
SOCKET
CON6
CON2
CON10
CON22
CON14
CON18
LED1
LED2
A
K
1kΩ
CON33
CON34
CON35
CON36

32-PIN TQFP PROGRAMMING ADAPTOR

SC ©2023

Fig.2: our four TQFP programming adaptors all follow this basic configuration, designed for maximum flexibility. The headers around the socket make it easy to connect any pin to GND, V_{CC}, V_{DD} or one of the pins on the programming header (CON35/CON36) via jumper wires. The optional regulators at the top can derive two different V_{DD} & V_{CC} supply rails from an external DC source.

While there are no bypass capacitors or anything like that, we've found it works fine as long as the jumper leads are kept short.

If you come across a different micro in the same package that uses a different pinout, rearranging the jumper wires to suit will be simple.

The socket is pretty expensive at about £30, including delivery, but it works well, and there aren't many better options.

If you only need to use it occasionally, a cheaper option is to design a PCB with an SOT-23-6 footprint wired to a programming header. Then, instead of soldering the chip to its footprint, simply hold it in place with a clothes peg or similar. That can work surprisingly well, but it's fiddly. We would only do that for the occasional chip; we wouldn't want to program dozens that way.

TQFP-32, -44, -48 & -64

We have two options for TQFP package chips. The first is the simplest but only suits chips like the ATmega328P. They are so common that you can get an adaptor that converts the pinout to the through-hole (DIP-28) equivalent. For example, this one costs £10, including delivery, at the time of writing: **pemag.au/link/abmv**

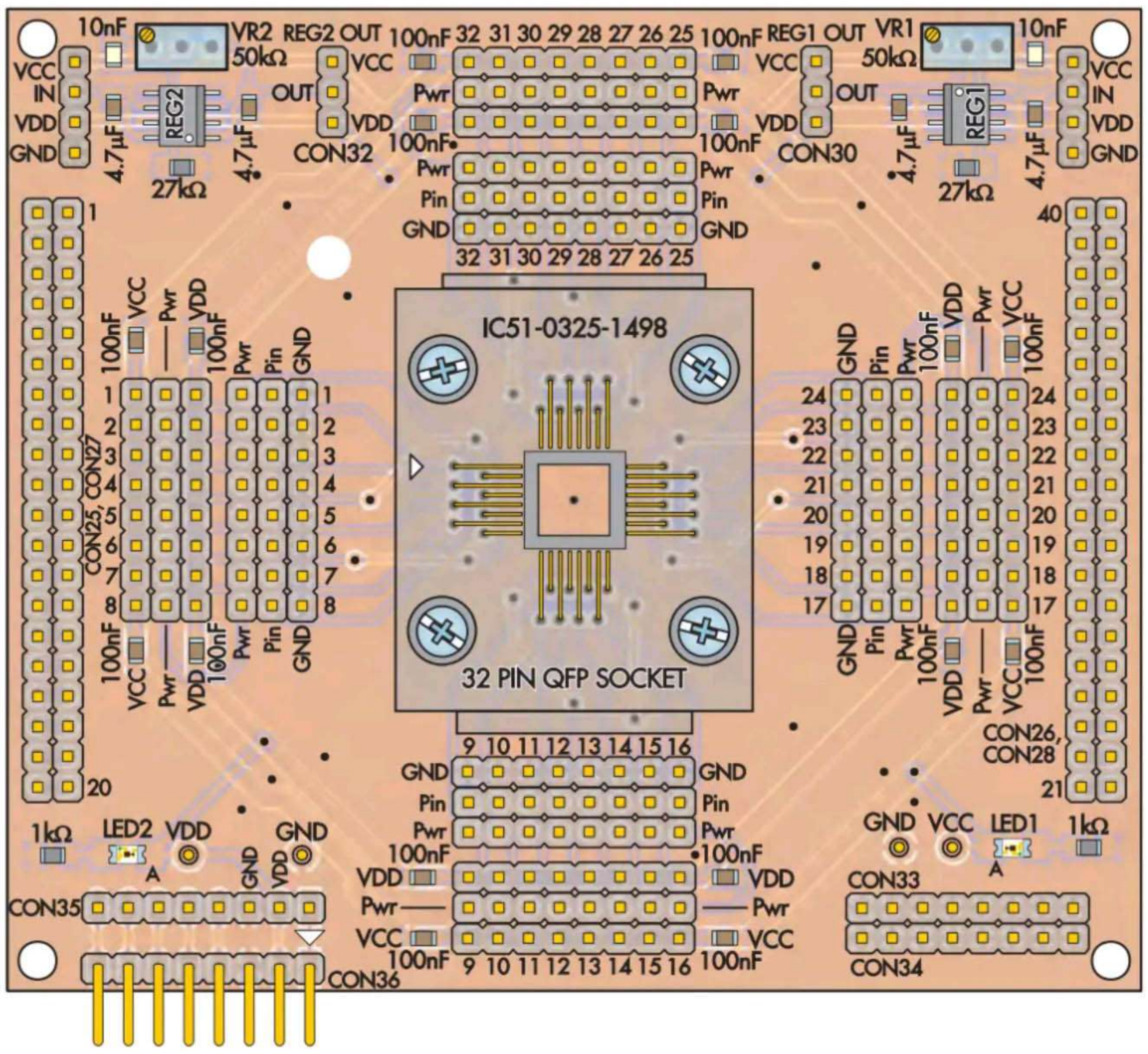


Fig.3: how the 32-pin TQFP programming adaptor rig would look if you fit all the components. That would give you all the options you need for programming any chip in this package, but in most cases, you can save yourself a bit of time and a few dollars by only adding the components you need for programming a given micro.

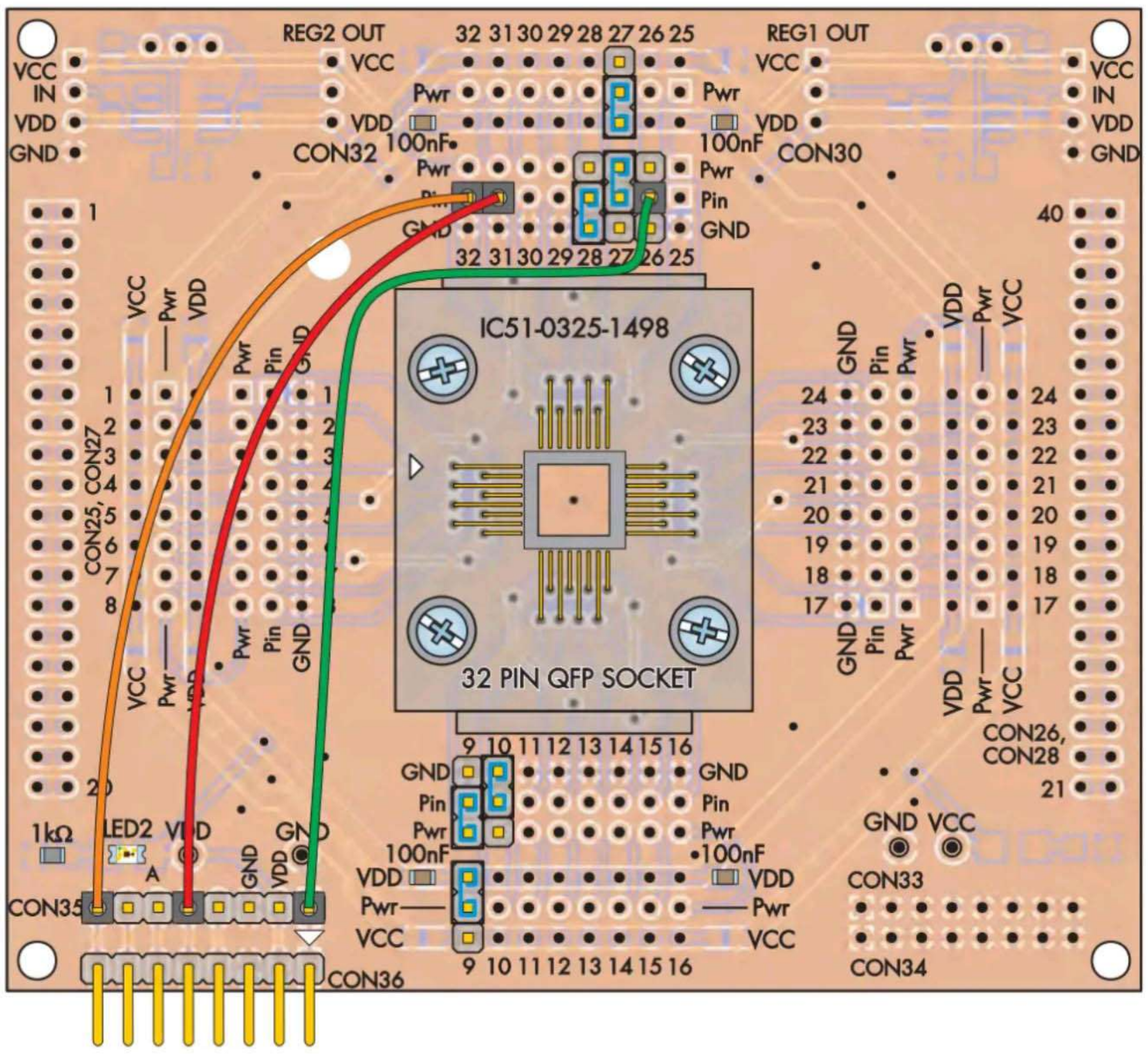


Fig.4: this shows how we built a very minimal programmer for the ATSAML10E16A-AUT ARM-based microcontroller from Atmel (now Microchip) on the same PCB shown in Fig.3. That chip is programmed using the AVR SWD (serial wire debugging) protocol, which uses different pins on the PICkit 4 8-pin header compared to PIC programming.

Say you have a means of programming a DIP ATmega328P, such as a TL866II or the newer T48 universal programmer. In that case, you just need to purchase the SMD adaptor, slot it into that programmer and away you go – see Photo 2.

A more flexible option that also works with chips like the ARM-based ATSAML10E16A mentioned above is our custom adaptor board that accepts a commercial TQFP programming/test socket. Its circuit is shown in Fig.2 and the matching PCB overlay in Fig.3. There are relatively few components on it, so it's pretty easy and inexpensive to build.

Our larger 44-pin, 48-pin and 64-pin adaptors follow the same pattern, so the following description will cover all of those. PCBs for all four versions are available from the SILICON CHIP Online Shop.

The test sockets are well made, have gold-plated contacts for a long life and only cost about £8-13 each. We have links to each one we've tested in the parts list. They have a staggered pin pattern unsuitable for use with protoboard and such, hence our custom PCB designs. Each PCB suits a specific socket.

Surrounding that socket, we have six rows of headers with one pin for each socket pin. The three closest to the socket allow you to use a jumper to connect that pin to GND or one of two power supply rails. You can also plug in a female header upside-down that lets you connect a capacitor from that pin to GND, or a capacitor to GND plus a connection to the supply rail.

For pins used for programming, it's a simple matter of fitting a short female-female jumper lead between the pin in the middle row and one of the ICSP header pins. Two sets of headers are provided to make it easy to connect these pins to an ICSP dongle like a PICkit or Snap programmer (PICkits can program AVRs now too).

The second set of three headers allows you to choose, for those pins connected to a power rail, which power rail that is. That is done by placing a jumper between the centre pin and either the V_{CC} or V_{DD} row. Note that most micros don't need two rails for programming, so you can use V_{DD} exclusively for those, but we wanted to provide maximum flexibility.

Each set of V_{CC} and V_{DD} pins along each side of the chip has a pair of

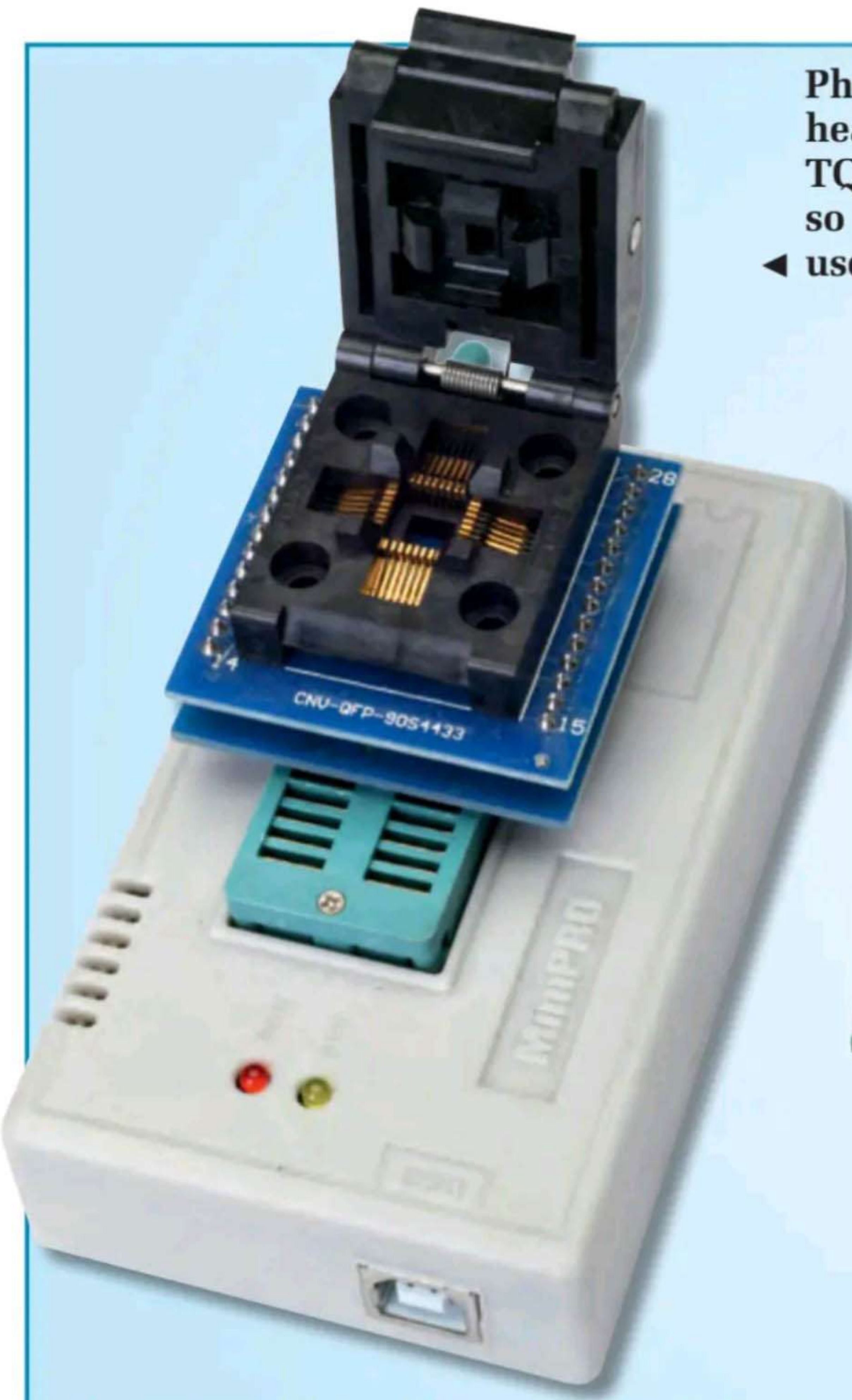

CNV-QFP-SOS4433
28
15
MiniPRO

Photo 2: this socket comes pre-mounted on a small PCB with a pair of SIL headers on the underside. They are routed to match up the pinout of the TQFP-32 version of the ATmega328 (and similar chips) to the DIP-28 version, so a standard DIP programer can be used to program the TQFP chips. ◄

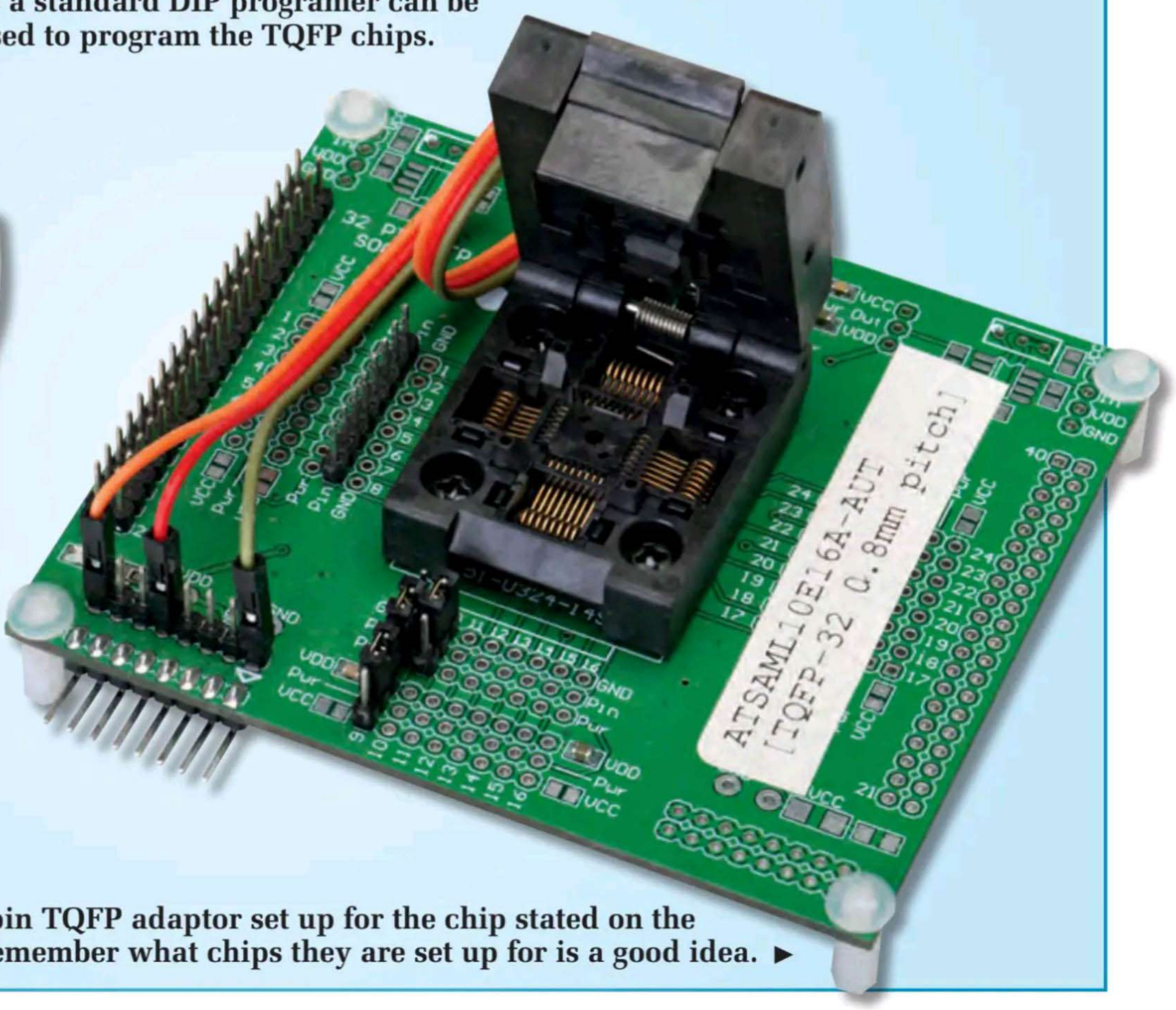

ATSAML10E16A-AUT
[TQFP-32 0.8mm pitch]

Photo 3: a minimalist assembly of the 32-pin TQFP adaptor set up for the chip stated on the label. Labelling the adaptors so you can remember what chips they are set up for is a good idea. ►

bypass capacitors to GND. This means you can get away without needing to add bypass capacitors closer to the chip in most cases; you can simply use jumpers to connect GND and V_{DD} where required.

Two adjustable or fixed linear regulators can be mounted in the board's upper left and upper right corners to supply either the V_{CC} or V_{DD} rail. In most cases, we use a PICkit to deliver power to V_{DD} via its header and don't bother with these. But again, this gives you flexibility. Using these regulators, you could derive V_{CC} and/or V_{DD} from USB 5V, a plugpack or a bench supply.

Pads are also provided to fit SMD LEDs to show when the V_{CC} and V_{DD} rails are powered.

Finally, there are rows of pairs of uncommitted pins that you can use for jumpering signals if required. The only connections on the board are between the pairs of pins.

Fig.4 shows the minimal parts needed to configure this board for programming the ATSAML10E16A, while Photo 3 shows our actual jig. That demonstrates that you only need to fit the parts you need for a particular application, and you can add more later if necessary. Here, we're using the PICkit 4 in its CORTEX SWD mode (it also supports AVR ISP, among other protocols).

One thing to note about these jigs is that the space around the socket is tight. That's because we've broken out the pins close to it, keeping the track lengths as short as possible. It's a little squashed, but we've programmed hundreds (if not thousands) of chips with these jigs and haven't had any real difficulty getting them in or out.

However, that could be a good reason to avoid fitting parts you don't think you'll need.

There are mounting holes in the corners for tapped spacers, so the jig sits flat on a bench. That makes them much easier to work with.

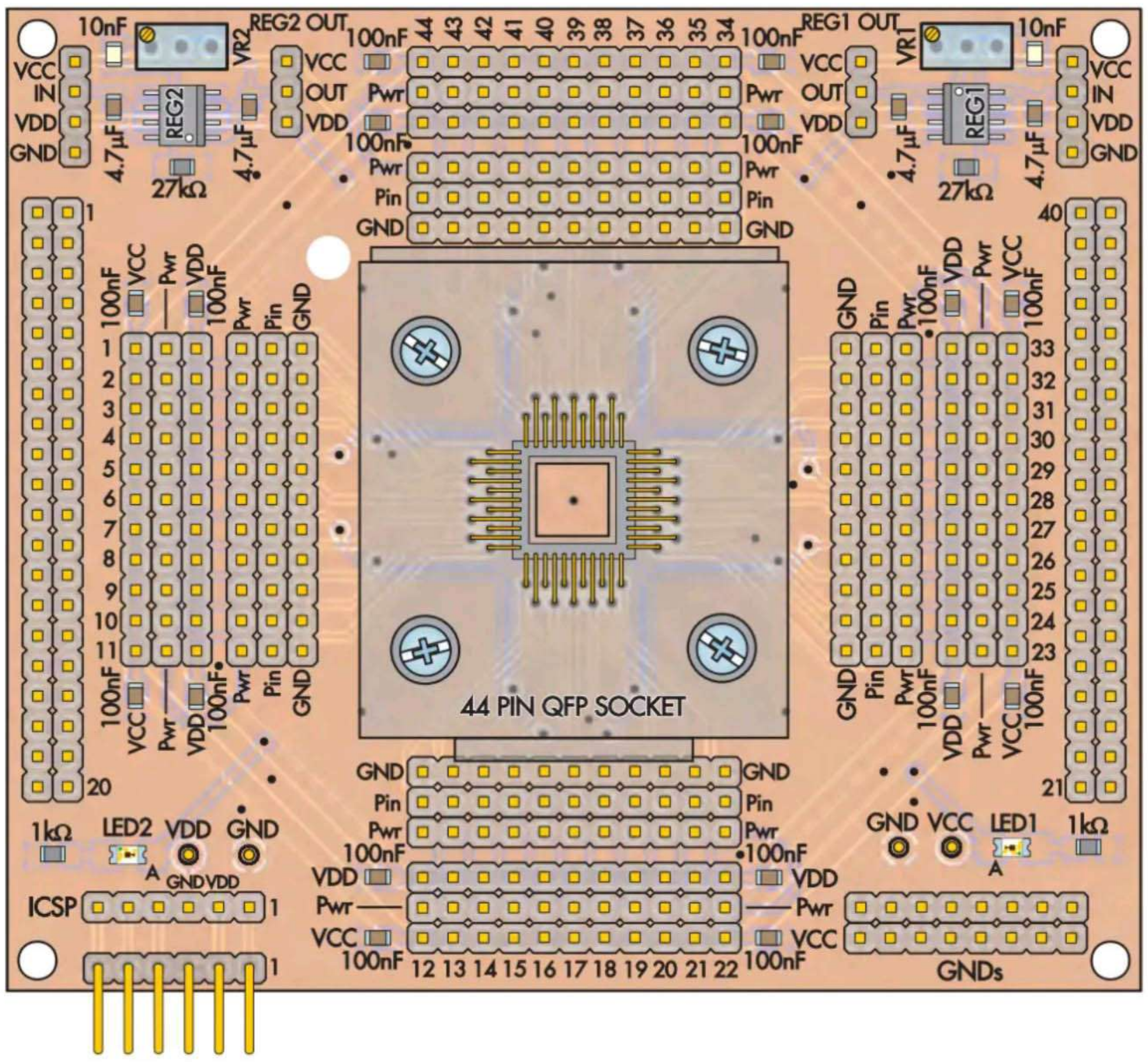

10nF
VCC
IN
VDD
GND
REG2
VR2
4.7µF
27kΩ
REG2 OUT
100nF
OUT
Pwr
Pin
44 43 42 41 40 39 38 37 36 35 34
REG1 OUT
REG1
VR1
1
40
1 2 3 4 5 6 7 8 9 10 11
20
33 32 31 30 29 28 27 26 25 24 23
21
44 PIN QFP SOCKET
1kΩ
LED2
LED1
A
GND VDD
ICSP
12 13 14 15 16 17 18 19 20 21 22
GNDs

Fig.5: here's where all the components go on the 44-pin TQFP programming adaptor. This package is pretty common for 8-bit PICs (eg, PIC16F18857 and PIC16F18877), 16-bit PICs (eg, dsPIC33-FJ128GP804), 32-bit PICs (eg, PIC32MX-170F256D-I/PT) and AVRs (eg, ATmega644PA).

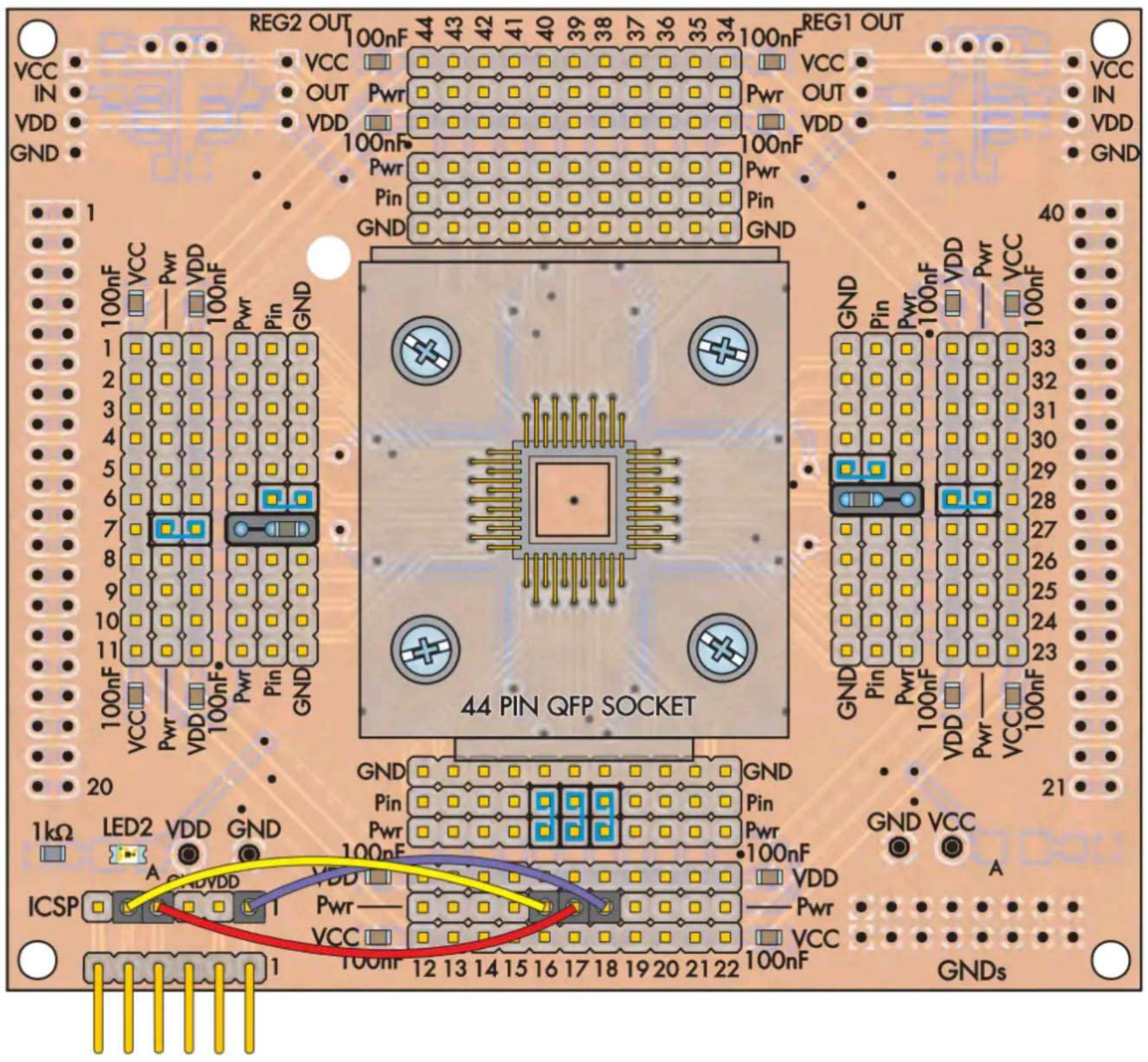


Fig.6: this shows how we wired up our assembled 44-pin TQFP programming jig to suit PIC16F18877-I/PT chips. Target power is delivered by the PICkit. The programming connections go via the pins on the "Pwr" row and then via jumpers to the IC pins to keep them away from the front socket opening.

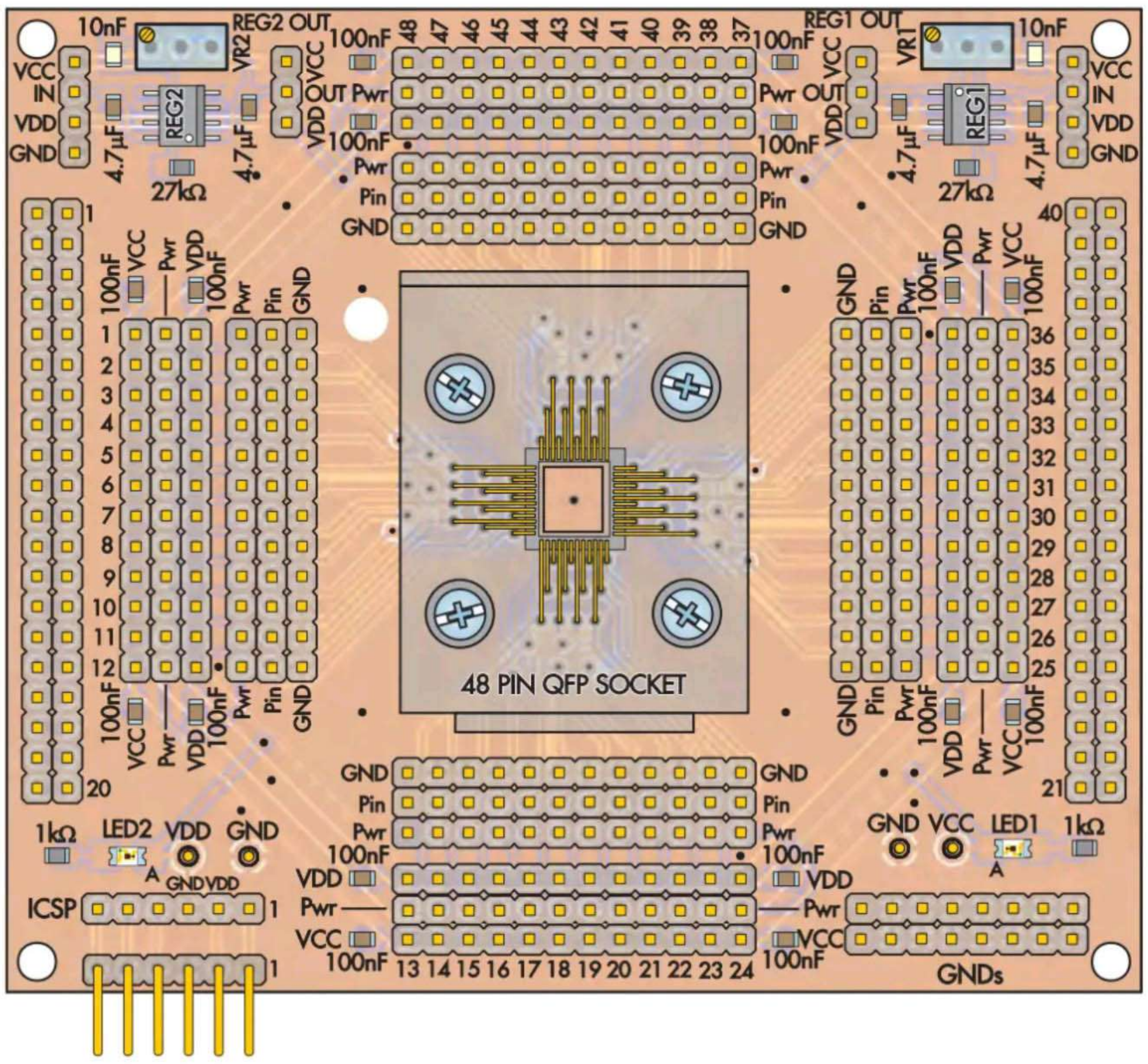
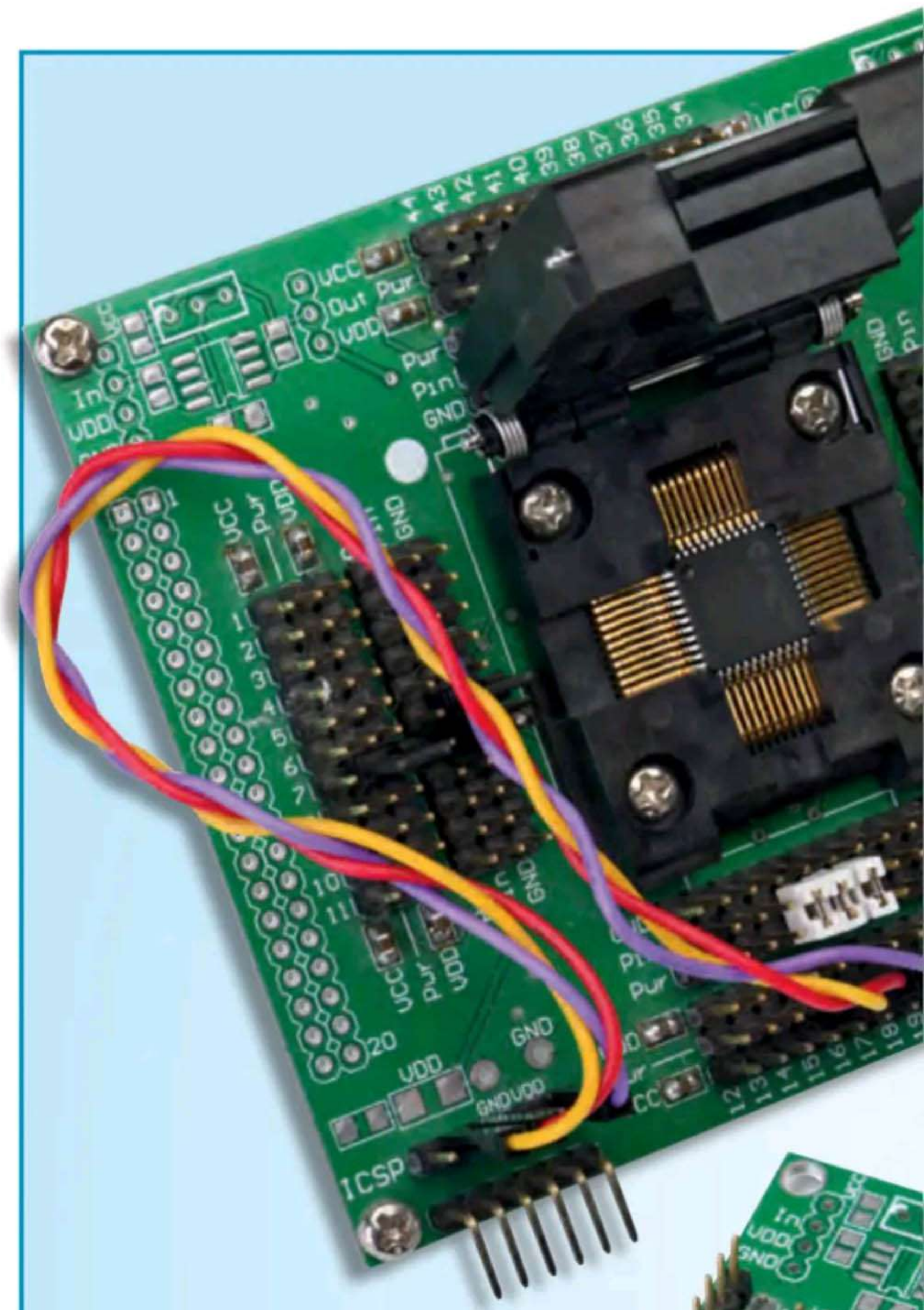


Fig.7: here's where the components go for the 48-pin TQFP Programming Adaptor. We have the parts to build one but haven't done so yet because micros in this package are far less common than either 44 pins or 64 pins.



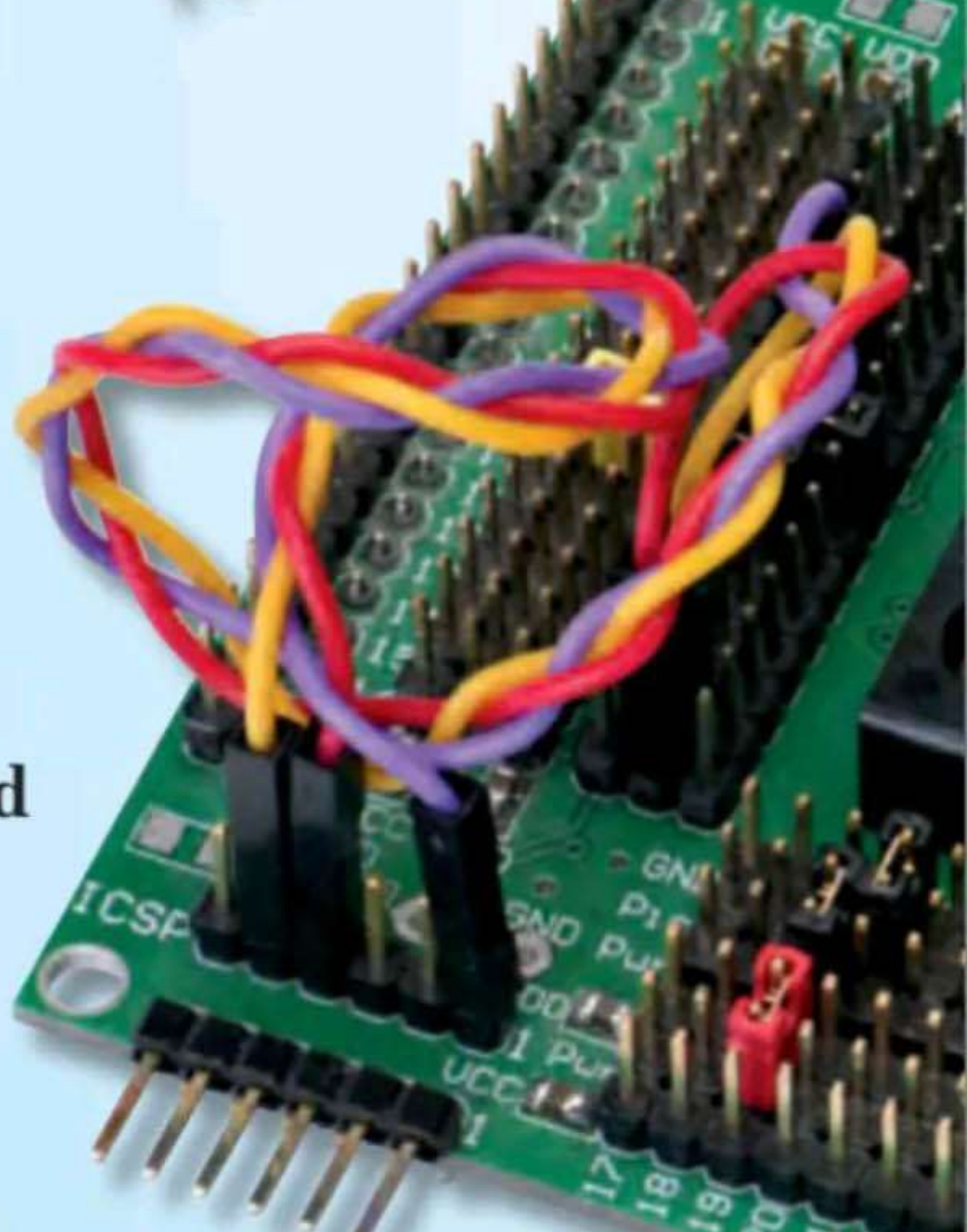


Photo 5: the 64-pin TQFP Adaptor board set up for a PIC32MX-470. If you don't need to make connections to the pins on the top row of the socket, especially in the middle, it can pay to leave those headers off, as it might allow you to open the socket clamshell wider, making it easier to get chips in and out.

We have placed a large filled circle on the PCB silkscreen at the upper-left corner of each TQFP socket to indicate where pin 1 of the IC would typically go. You then line up the dot or divot on the chip with that marking.

All that means is that the pin number labelling on the headers will be correct when the chip is orientated like that. You could use a different orientation and reroute the connections to suit if you wanted to, as there are no fixed connections to the socket on the board. Still, keeping pin 1 in the upper left-hand corner is less confusing.

TQFP-44 programming rig

We haven't drawn the circuit for this one as it's the same as Fig.2 but expanded for the extra pins. The PCB overlay is shown in Fig.5. Photo 4 shows our jig, currently configured for the PIC16F18877-I/PT, while Fig.6 shows that wiring. Other chips we've programmed with this rig include the PIC32MX170F256D-I/PT and ATmega644PA.

Note in Photo 4 how we've plugged a 3-pin socket into the header for pin 28 (V_{DD}) with a capacitor soldered across one pair of pins and a short wire across the others. It acts as a jumper to connect V_{DD} to that pin while providing a bypass capacitor to GND.

TQFP-48 programming rig

Again, the circuit is the same as Fig.2 but with extra pins on the socket, while the PCB overlay is shown in Fig.7. We haven't built one yet as we don't need it. That's because most microcontrollers in 48-pin TQFP packages are ARM-based types that we haven't used (from Infineon, Renesas or Silicon Labs).

Still, we designed the board and

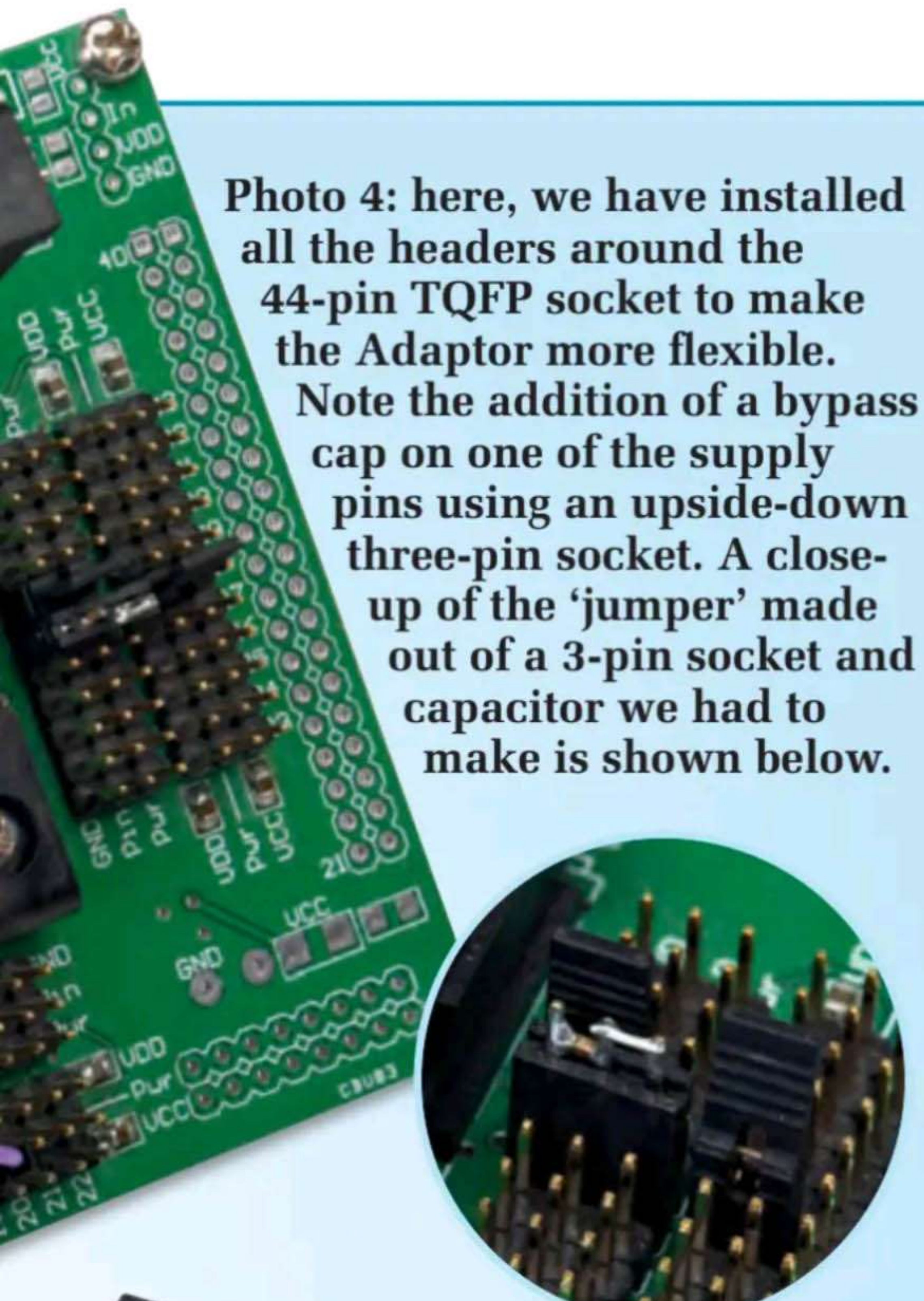
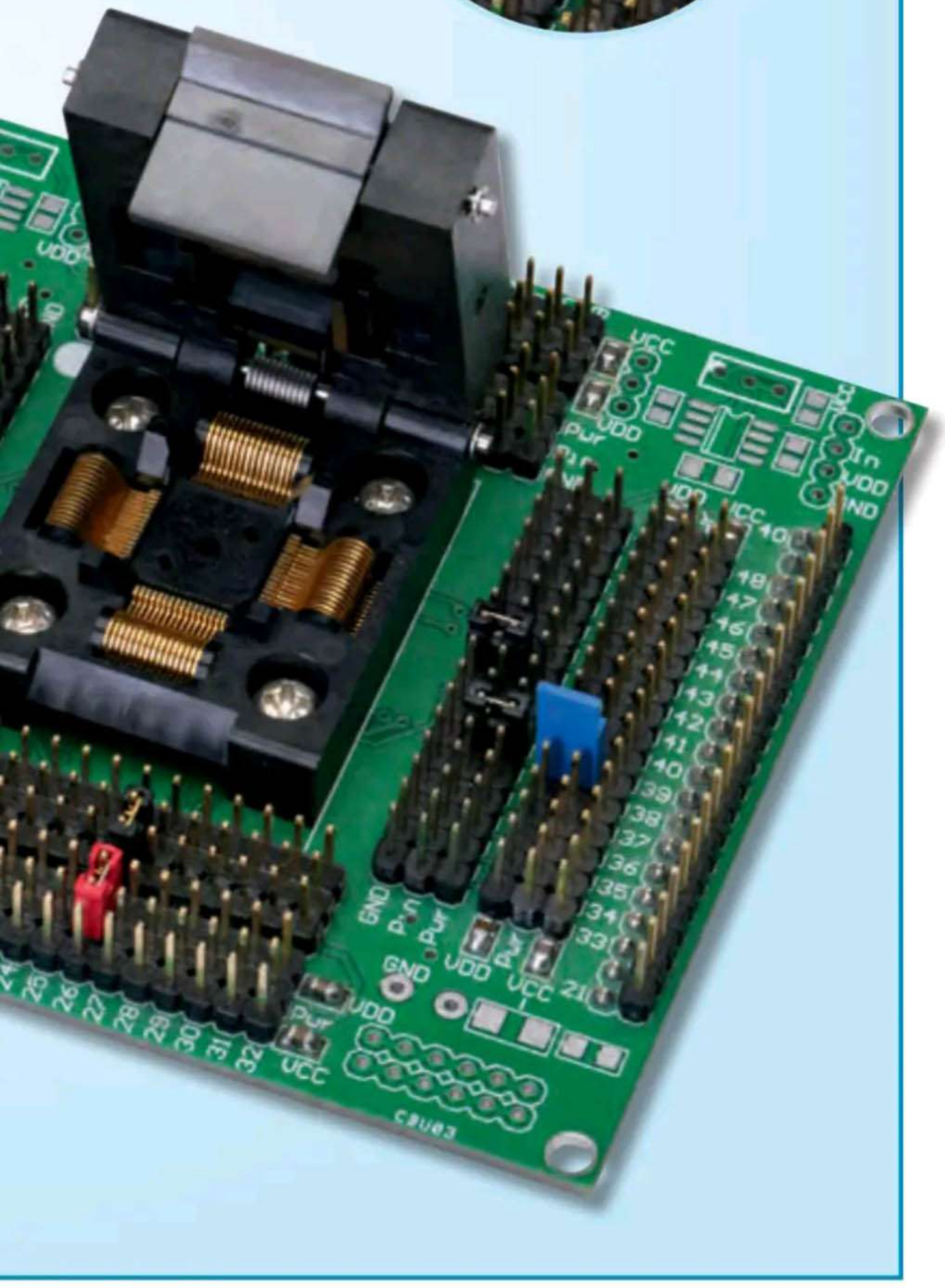


Photo 4: here, we have installed all the headers around the 44-pin TQFP socket to make the Adaptor more flexible. Note the addition of a bypass cap on one of the supply pins using an upside-down three-pin socket. A close-up of the 'jumper' made out of a 3-pin socket and capacitor we had to make is shown below.



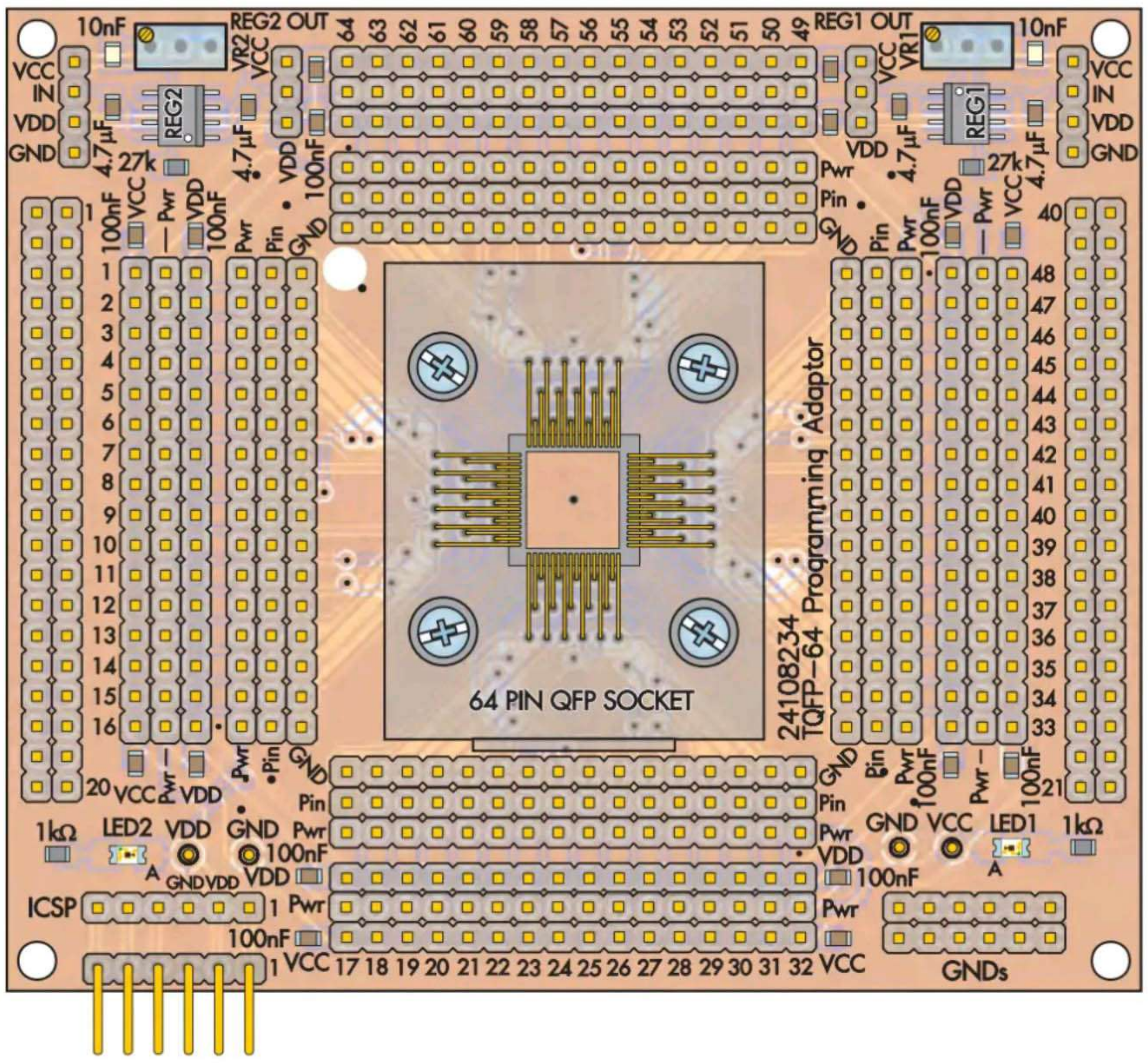

REG2 OUT
REG1 OUT
10nF
4.7μF
27k
100nF
VCC
IN
VDD
GND
Pwr
Pin
ICSP
1kΩ
LED2
LED1
64 PIN QFP SOCKET
24108234 TQFP-64 Programming Adaptor
GNDs

Fig.8: this is the 64-pin version of the TQFP Programming Adaptor. It's a reasonably common package, especially for 32-bit PIC microcontrollers and some PIC16s, PIC18s, PIC24s, dsPIC33s, Atmel ATSAM chips and more. This is basically the same as the other boards but with more socket and header pins.

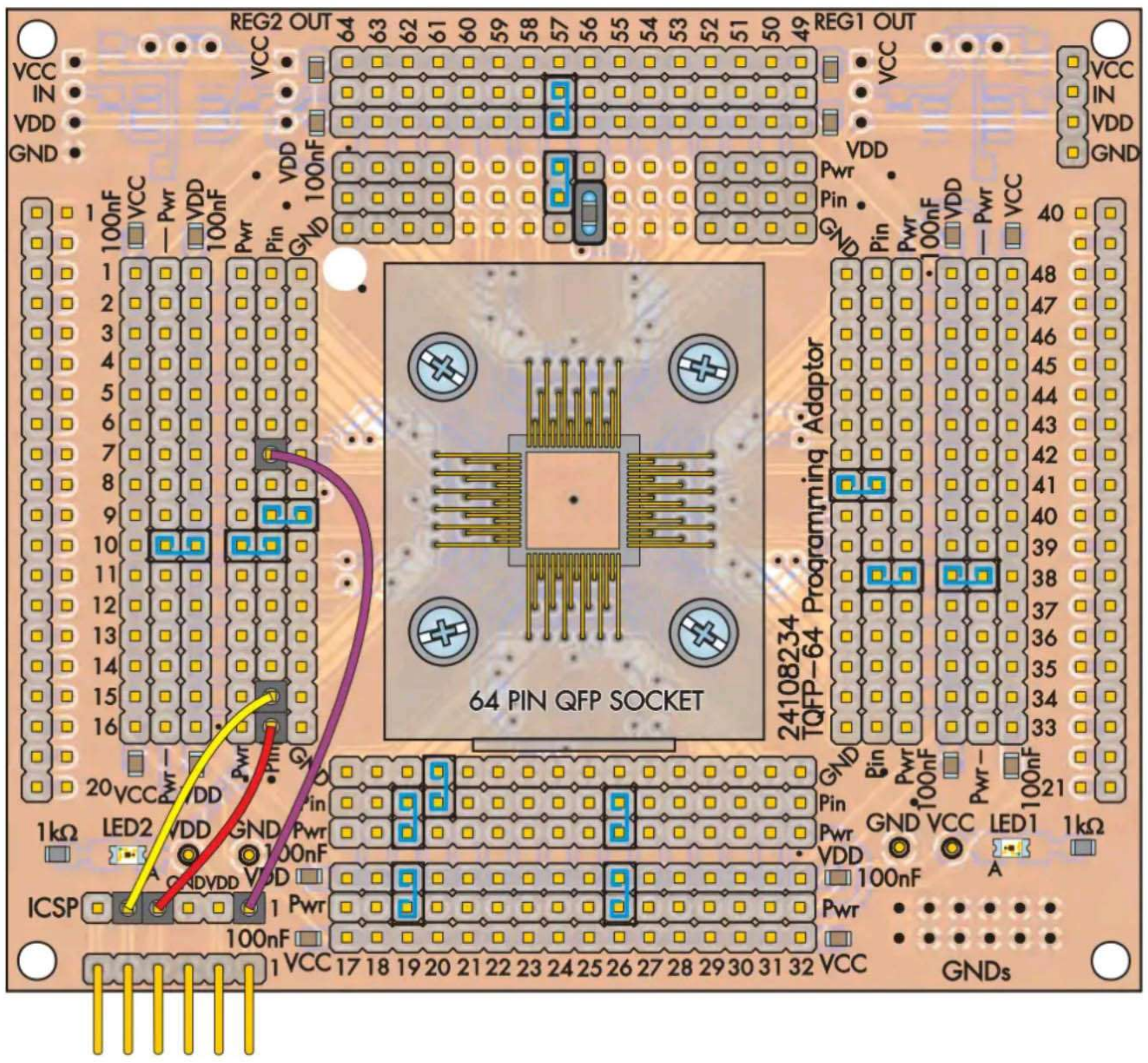

REG2 OUT
REG1 OUT
VCC
IN
VDD
GND
Pwr
Pin
ICSP
1kΩ
LED2
LED1
64 PIN QFP SOCKET
24108234 TQFP-64 Programming Adaptor
GNDs

Fig.9: the 64-pin Programming Adaptor set up for the PIC32MX-470F512H-120/PT used in the Micromite Plus.

We removed the plastic from some headers for pins 53-55 and 58-60 so they would sit lower and give more clearance for the TQFP socket hinges. They're still long enough to fit jumpers.

sourced a socket, so we decided to make the PCB available to anyone who might need it.

TQFP-64 programming rig

Again, this circuit is simply that of Fig.2 but with twice as many socket pins and associated header pins. The PCB overlay is shown in Fig.8, with Fig.9 being the minimal configuration for programming a PIC32MX470F512H-120/PT or similar (this should also suit 64-pin dsPICs). Photo 5 shows our jig for programming the PIC32MX470F512H-120/PT.

We also built a TQFP-64 programming jig for the powerful PIC32MZ2048EFH064-I/PT chip that Phil Prosser likes to use in his projects. As shown in Photo 6, we didn't use our custom board for this but instead purchased a socket that came on a PCB with 16-pin headers on all four sides (on the underside).

We then mounted that on a protoboard via four 16-pin sockets and soldered the bypass capacitors and programming header to that. Connecting the pins to GND, V_{DD} and the programmer pins was done by point-to-point wiring using Kynar (wire wrap wire), which is thin but stiff and easy to work with.

We could have used the custom jig in this role, but we wanted to try a different approach. It was a little work to do this but it worked fine. At the time of writing, this adaptor costs £16, including delivery and can be purchased from pemag.au/link/abmw

TQFP-100 programming rig

We haven't bothered to make a custom 100-pin board for several

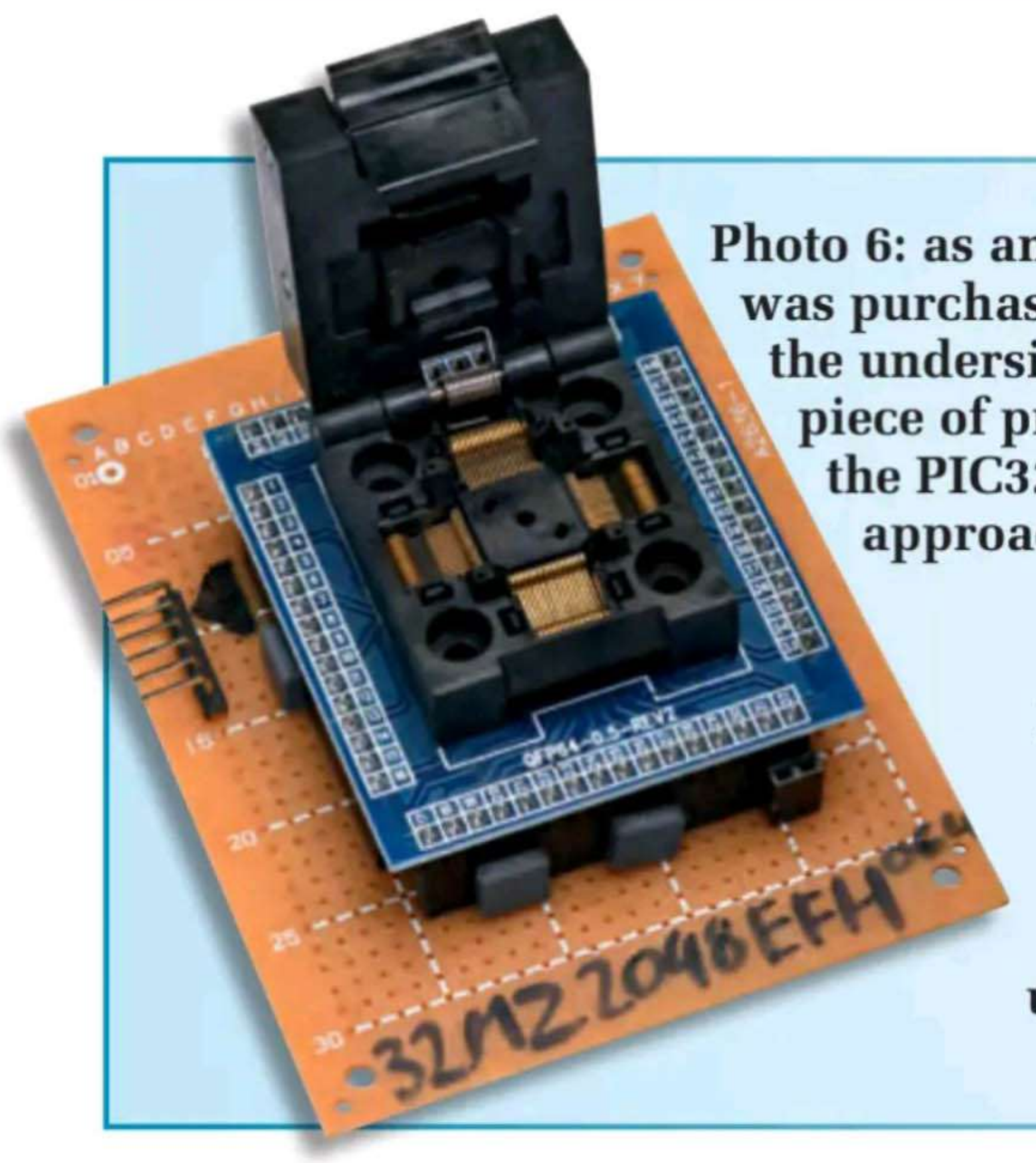


Photo 6: as an alternative approach, this 64-pin TQFP socket was purchased already fitted to a board with headers on the underside. We then soldered matching sockets on a piece of protoboard and hardwired a programmer for the PIC32MZ. It's a bit less flexible than the other approach, but it works.

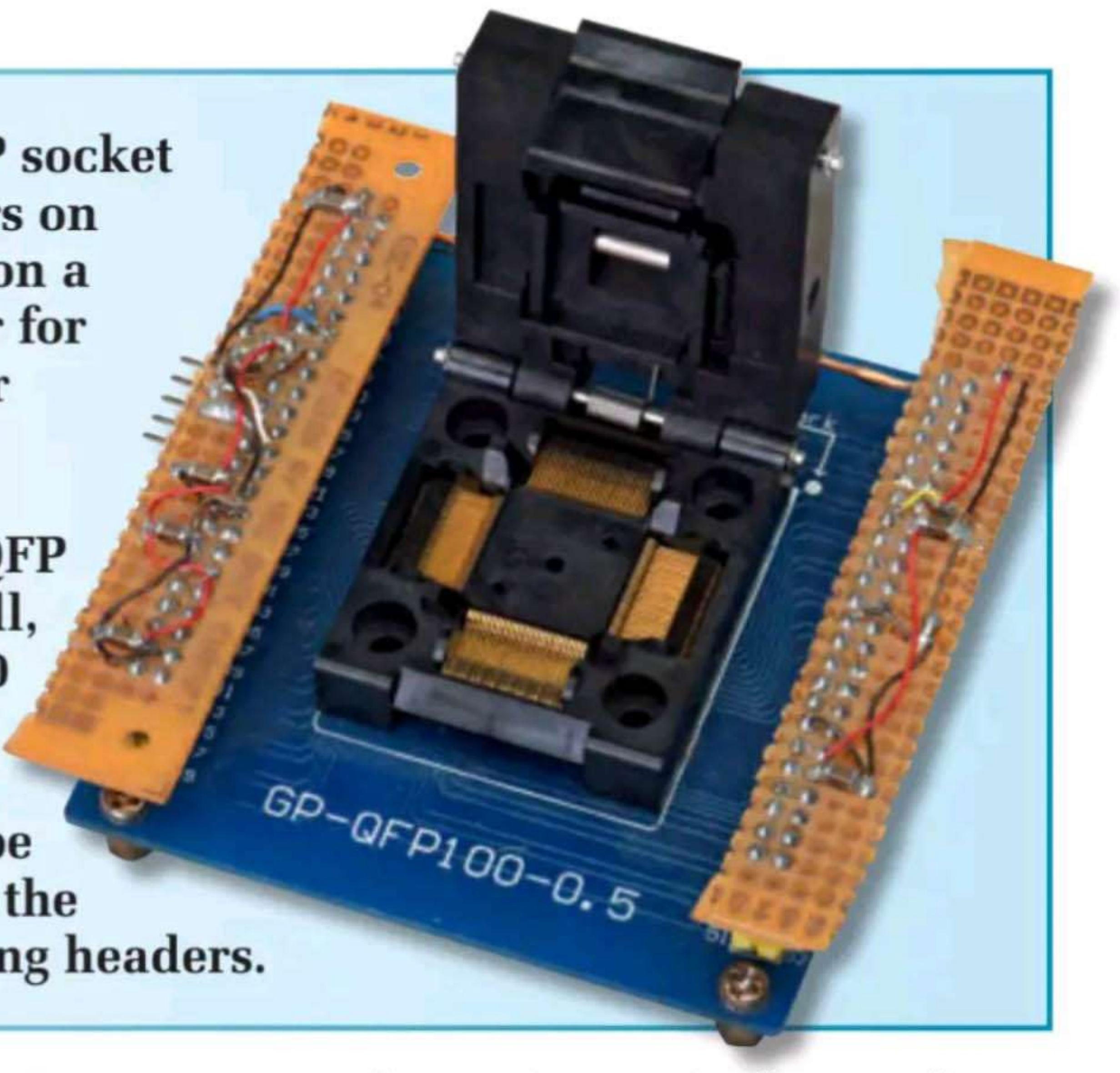


Photo 7: this hand-made 100-pin TQFP programming adaptor has served us well, programming all the PIC32MX470 chips for the Explore 100. The added protoboards (joined by two wires across the back, for V_{DD} & GND) can be unplugged as they are on sockets that fit the pre-existing headers.

reasons. The main one is that we have programmed only one 100-pin micro to date, the PIC32MX470F512L-120/PF used for the Micromite Plus Explore 100. So we only needed a simple fixed jig.

We were able to buy a 100-pin socket pre-mounted on a PCB with two pairs of 50-pin DIL headers. It was relatively easy to solder strips of protoboard to these headers and then use that to add bypass capacitors, a programming header and the connections necessary for programming, as shown in Photo 7. It's a little dusty, but it works!

As with the PIC32MZ rig, the protoboard is attached via sockets, so we could theoretically unplug them and change the socket to work for a different micro if we need to. Note the two wires running across the back of the socket that join V_{DD} and GND between the two sides. The programming header is on the underside of the left-hand board and is just visible in the photo. The point-to-point wiring was done using Kynar wire-wrap wire, as it's easy to work with.

This adaptor costs £26, including delivery at the time of writing, and can be purchased from pemag.au/link/abmx

Conclusion

These programming rigs are somewhat specialised, but they certainly come in handy when you need them. You might build one when embarking on a project that uses a particular chip, and you could build up a collection over time as you work with micros in different packages.

We put some effort into creating these PCBs to make them flexible and easy to work with. So any readers with similar needs would benefit from being able to use our PCBs and follow the same general strategies. PE

Parts List – TQFP Programming Adaptors

Parts required for all versions
1 6- or 8-pin header, 2.54mm pitch
1 6- or 8-pin right-angle header, 2.54mm pitch
16 M2012/0805 100nF 50V X7R ceramic capacitors
10 small jumper shunts
3 short female-female jumper wires
4 M3 tapped spacers
8 M3 × 6mm panhead machine screws
4 M3 hex nuts
1 serial programmer to suit chip(s) being programmed

TQFP-32 programming adaptor
1 double-sided PCB coded 24108231, 95 × 82.5mm
1 TQFP-32 'clamshell' test/programming socket [pemag.au/link/abmy]
24 8-pin headers, 2.54mm pitch

TQFP-44 programming adaptor
1 double-sided PCB coded 24108232, 95 × 82.5mm
1 TQFP-44 'clamshell' test/programming socket [pemag.au/link/abmz]
24 11-pin headers, 2.54mm pitch

TQFP-48 programming adaptor
1 double-sided PCB coded 24108233, 95 × 82.5mm
1 TQFP-48 clamshell test/programming socket [pemag.au/link/abn0]
24 12-pin headers, 2.54mm pitch

TQFP-64 programming adaptor
1 double-sided PCB coded 24108234, 95 × 82.5mm
1 TQFP-64 test/programming socket [pemag.au/link/abn1]
24 16-pin headers, 2.54mm pitch

Optional parts for all boards
1 or 2 M3216/1206 SMD LEDs plus 1kΩ M2012 resistors (for V_{DD}/V_{CC} indication)
1 two-pin female header plus M2012/0805 22µF 6.3V X5R capacitor (for micro pins that need a capacitor to GND)
1+ three-pin female header plus M2012/0805 100nF 50V X7R capacitor (for micro pins that need local bypassing)
1 2×8 pin DIL header, 2.54mm pitch (for extra GND terminals)
2 2×20pin DIL header, 2.54mm pitch (for extra connecting terminals)

Extra parts per onboard regulator (up to two regulators per board)
1 LP2951D adjustable linear regulator, SOIC-8 (REG1/REG2)
1 50kΩ top-adjust multi-turn trimpot (VR1/VR2)
1 27kΩ 1% M3216/1206 or M2012/0805 SMD resistor
2 4.7µF 25V X5R M2012/0805 SMD ceramic capacitors
1 10nF 50V X7R M2012/0805 SMD ceramic capacitor
1 4-pin header, 2.54mm pitch
1 3-pin header, 2.54mm pitch
1 jumper shunt

AOShop

The home for specialist audio, analogue and historic components – provided by Jake Rothman, *PE's Audio Out* columnist. The AOShop is your best bet for classic analogue 'NOS' (new old stock) components, including all parts for *Audio Out* projects and designs.

Ge semiconductors

Small-signal PNP transistors

NKT214F, OC57, OC59, 2N1377, 2N525	**£1.00**

Low-noise PNP transistors

GET106	**£2.50**

Small power PNP transistors

AC153, AC153K, AC188	**£1.50**

Small power NPN transistors

AC176, AC141K	**£2.00**

High-voltage PNP transistors

OC77, CV7001	**£1.50**

TO3 PNP power transistors

OC22, CV7054 (OC23), OC25, OC35, OC36, AD143, AD149, AD161, AD162	**£2.00**
AD140	**£3.50**
AD149, AD161/2 matched pair	**£5.00**

RF PNP transistors

OC41, OC42, 2SA12, 2SA53,AF124, AF178, GET872A	**£1.50**

NPN transistors

OC139, OC140, ASY73	**£2.00**

Diodes

CV7049 (OA10), CG92 (OA91)	**£0.50**

Si semiconductors

Diodes

ZC5800 RF Schottky	**£0.20**

Low-noise silicon transistors

BFW16A, 2SC3071, 2SC3068, 2SA1016K, 2SC2362K, 2SA970BL	**£1.50**
2SC2204, 2SD655, BC550C	**£0.50**
ZTX651	**£0.30**
ZTX751	**£0.50**

RF transistors (suitable for Theremin)

BF199	**£0.50**

Audio power MOSFETs

Exicon 10N20, 10P20	**£6.50**
Hitachi 2SJ99, 2SK343	**£3.50**
Hitachi 2SJ56, 2SK176	**£8.50**

JFETs

BFW11, BFW10, TIS73L, J177, J113, U1994, U1898, 2SJ176, J201	**£1.00**
J175, J176, J112, J111, 2N3820, 2N5467, BF244, 2N5460, J230	**£0.60**

Small power output/driver transistors

2SB649A, D669A, 2SA1208, 2SC2910, MJE253G, MJE243G, 2SA1725, 2SC4511	**£1.20**
BD139,BD140, BD135, BD156, BD435, BD436, MPSA63, BCV46	**£0.50**

MOSFETs

ZVP2106A	**£0.30**
Dual-gate 3SK45, BFS28	**£1.50**

Metal-cased transistors

BC143	**£0.35**
2N1711	**£0.50**
BCY71	**£0.30**
BC109C	**£0.60**

Amplifiers

LM318 high-speed op amp	**£0.35**
µA709 metal-cased op amp	**£2.00**
5534H metal-cased op amp	**£3.00**
LM384, TDA2030A, TDA2050V	**£1.50**
TAA435 (Mullard power amp driver for AD161/AD162)	**£1.20**

Synthesiser ICs

That/dbx 2180 VCA/VCF	**£6.00**
CA3080 VCA/VCF (vocoder)	**£2.80**
CA3280 VCA/VCF	**£4.00**
LM13600/ LM13700 VCA/VCF	**£2.00**
CA3086 transistor array	**£1.00**
PT2399 echo/delay	**£1.50**

Dual transistors

2N2639, 2N2223, 2N2910 (NPN)	**£4.00**
E401 (JFET Moog)	**£4.00**
2SK2145-Y dual JFET	**£0.80**
2N5564 JFET	**£8.00**
HN3C51F, HN3A51F	**£1.00**
DMMT3904/6, HN1A01F, HN1C01F	**£0.50**

Loudspeakers

PE Mini-Monitor

Volt PE165 6.5-inch woofer (each)	**£85**
Morel MDT29 tweeter (each)	**£25**
Kit *pair* of PE165/MDT29 plus Volt crossover parts and PCBs	**£299**
Monacor DT-28N tweeter (each)	**£35**
Vifa 19mm BC20SC15-04 tweeter (each)	**£15**

Volt crossover inductors

1.2mH, 1.5mH, 2mH, 2.7mH, 0.5mH (tapped at 0.3mH) (each)	**£5.85**

Fully assembled and tested high-quality speaker prototypes – ask for details

LS3/5As and other similar speaker systems (pair)	**£200-£350**

Fully tested reclaimed speakers

Vifa BC14 5-inch woofer (each)	**£10**
Vifa TC26 1-inch tweeter (each)	**£10**

Low-price speakers

Philips 4-inch 4070	**£2.00**
EMI 10x6-inch, 30Ω Alnico	**£7**
64mm 64Ω neodymium	**£1.20**

Capacitors

Note '10/63' denotes '10µF 63V'.

Polyester

3.3/100, 4.7/250, 4.7/63	**£1.00**
5.6/63, 8.2/63, 10/63	**£2.00**
Mullard 'Mustard' C296 0.22/400	**£2.00**

Polycarbonate

Axial 2.2/63 1%, 4.7/160, 6.8/63	**£1.00**

Radial

6.8/160V, 10/63	**£2.00**
Reclaimed 22/63	**£2.00**

Polystyrene

Philips 1% 4.7nF/160, 6.2nF/500, 12nF/63, 22nF/63, 110nF/63, 24nF, 2nF	**£1.00**
RIFA 1% 100nF/100, Suflex 90.9nF 0.5%	**£2.00**
Suflex 2.5% 10nF/63 (rad. or ax.)	**£0.50**

Radiation resistant

Siemens cellulose acetate MKL 2.2/25	**£0.80**

Electrolytic – Mullard blue 017 series

10/25, 22/25, 100/10	**£0.50**
150/40, 470/40, 1000/40	**£1.00**

Tantalum – axial metal cased

22/50, 47/35, 68/25, 100/20, 120/10, 150/16, 220/10, 330/6	**£2.00**
22/35, 33/35, 47/20, 68/15, 100/10, 150/6	**£1.25**
4.7/50, 6.8/35, 10/25, 10/35, 22/15	**£1.00**

Axial moulded-case tantalum

Kemet axial 6.8/10	**£0.30**
Kemet radial 33/10	**£0.40**
STC radial 100/20	**£1.50**

Tantalum bead

22/50	**£1.00**
470/3	**£2.00**
680/6.3	**£3.00**

Wet tantalum

220/25 axial	**£3.20**
Castanet button 140/30, 470/3	**£3.20**
Hughes 540/10	**£3.20**

Bipolar

Hermetic bipolar tantalum 16/35	**£3.20**
Elcap axial 10/50	**£0.50**
Generic radial 100/16, 470/35, 100/ 63, 22/35, 4.7/35, 220/16	**£0.50**

Philips solid-aluminium (axial) 121/123

47/16	**£1.00**
330/6.3	**£2.00**
100/35	**£4.00**

Philips Pearl 122 series (radial)

10/16	**£0.30**

Silvered mica (radial)

1nF/500 1%	**£1.00**

Trimmer capacitors

Vishay plastic-film 4-40pF	**£0.80**
Vishay plastic-film 5.5-45pF	**£1.00**
Vishay plastic-film 5-80pF	**£1.20**
Vishay plastic-film 10-250pF	**£2.00**
Mica 1-12pF, 2-40pF	**£1.00**

Audio transformers and inductors

Eagle transformers

LT44, LT722 driver, LT700, LT723 500Ω output	**£2.50**
LT30 500mW output	**£3.50**
5:1 interstage	**£1.50**

Repanco

T/T3 splitter transformer	**£4.00**
CH2 5mH RFC	**£2.00**

Balanced output transformer

Vigortronix 600Ω VTX-101-007	**£10**
Vigortronix 600Ω VTX-101-3001	**£10**
Vigortronix 600Ω VTX-101-3002	**£15**
Gardners 150Ω	**£10**
Reclaimed BBC LL74/MPC nickel core 600Ω	**£12**
Low-level 1:1 600Ω	**£4**
Reclaimed mic input transformer	**£15**

Inductors

82µH, 4.7mH, 100µH, 270µH, 10µH, 14µH (low Z)	**£0.50**
7-inch ferrite rod with MW and LW windings	**£3.00**

Special resistors

Bourns wire-wound trimmer 5kΩ 3059 JM panel-mount	**£2.00**
Thermistor RA53, R13	**£4.00**
A13	**£2.00**
Thermistor CZ1, CZ6	**£1.50**
Holco H2 2.2MΩ 1W, 1%	**£1.00**
Welwyn 1GΩ 2W	**£1.00**
5k Bourns 3321H cermet trimmer	**£0.50**

Potentiometers

Bourns 81

25kΩ lin cermet	**£2.00**
25kΩ lin, 5kΩ lin conductive plastic	**£3.00**

Bourns 91

10k dual-gang lin or log	**£10.00**

Plessey moulded-track

5kΩ log with switch	**£3.00**
50kΩ A/log	**£3.00**
Mil 250kΩ lin dual	**£5.00**

Alpha

16mm 4.7kΩ A/log	**£0.80**
220kΩ A/log	**£0.80**
10kΩ lin centre-detent dual-gang	**£1.50**

Allen Bradley J series/Honeywell

10kΩ lin 1 million cycles	**£5.00**
Blore Edwards AB 45 dual 5kΩ A/log with switch	**£3.50**
Alps RK9 dual-gang 5kΩ RD law	**£4.00**
Alps 50k log tapped motorised stereo	**£5.00**
BI P260 500kΩ log conductive plastic 1 million cycles	**£2.00**
Colvern wire-wound 100kΩ or 50kΩ dual-gang 3W	**£5.00**
Mil Spec hermetic 10Ω	**£8.00**

Miscellaneous

PCBs

Pocket Theremin (*EPE*, 1996)	**£2.00**
Elysian Theremin (*EPE*, 1996)	**£6**
Synth VCF, VCO (*EPE*, 2017)	**£3.00**
48V PSU (*EPE*, 2019)	**£3.00**
SMT dual transistor adapter	**£0.40**

Relays

STC hermetic 4190 24V DPDT	**£4.00**

Switches Blore Edwards/Plessey

MH1 4-pole 3-way	**£3.00**
Elma 01 2-pole 6-way	**£6.00**
3PDT guitar pedal stomp switch	**£2.50**

Valves

EC88, PCL84	**£3.00**

Contact

Jake Rothman
The Old Rectory, Arlais Road,
Llandrindod Wells, Powys
LD1 5HE (visit by appointment)

+44 (0)1597 829102

jrothman1962@gmail.com

Minimum order £5.00 inc post
Quantity discounts negotiable

Payment

PayPal, bank transfer, cheques (payable to "J Rothman", UK pounds only)
No VAT payable

Postage

Small Jiffy bag £2.99
Small package £4.99
Big boxes and overseas at cost – ask for a quote

AUDIO OUT

AUDIO OUT
L
R

By Jake Rothman

Audio switching Part 5 – more on electronic switching

We pick up here where we left off last month, describing various practical audio switching circuits implemented using solid-state devices like JFETs or ICs, rather than mechanical relays and such. They have various advantages, such as silent operation, reduced cost and lower power consumption.

Capacitor coupling

When the FET is switched off with a negative control voltage, a DC path to ground is necessary for the bootstrap resistor, which can be the low impedance of an op amp output. Without this path, it won't switch off.

If a coupling capacitor is used to block the DC offset of the stage feeding the JFET to prevent clicks, this path is blocked. Douglas Self connects the bootstrap resistor before the capacitor to give the required path, as shown in Fig.83.

Solid-state logic (SSL)

Back in January 1989, I took the SSL maintenance course in Begbroke, Oxfordshire. This was for the massive £250,000 G-series mixing consoles that every top studio then had to have. One of the lecturers was SSL's principal analog design team leader, Andy Millar, who explained the subtleties of FET switching.

His team's work led to the development of the SSM2402 audio switch at Analog Devices/Precision Monolithics Inc. In this circuit (Fig.84[a]), the bootstrapping was done by an op amp, which ensured that V_{GS} was zero while allowing the gate control signal to pass through. The control input to the op amp was via a current source; otherwise, the follower would become an amplifier boosting the bootstrap signal.

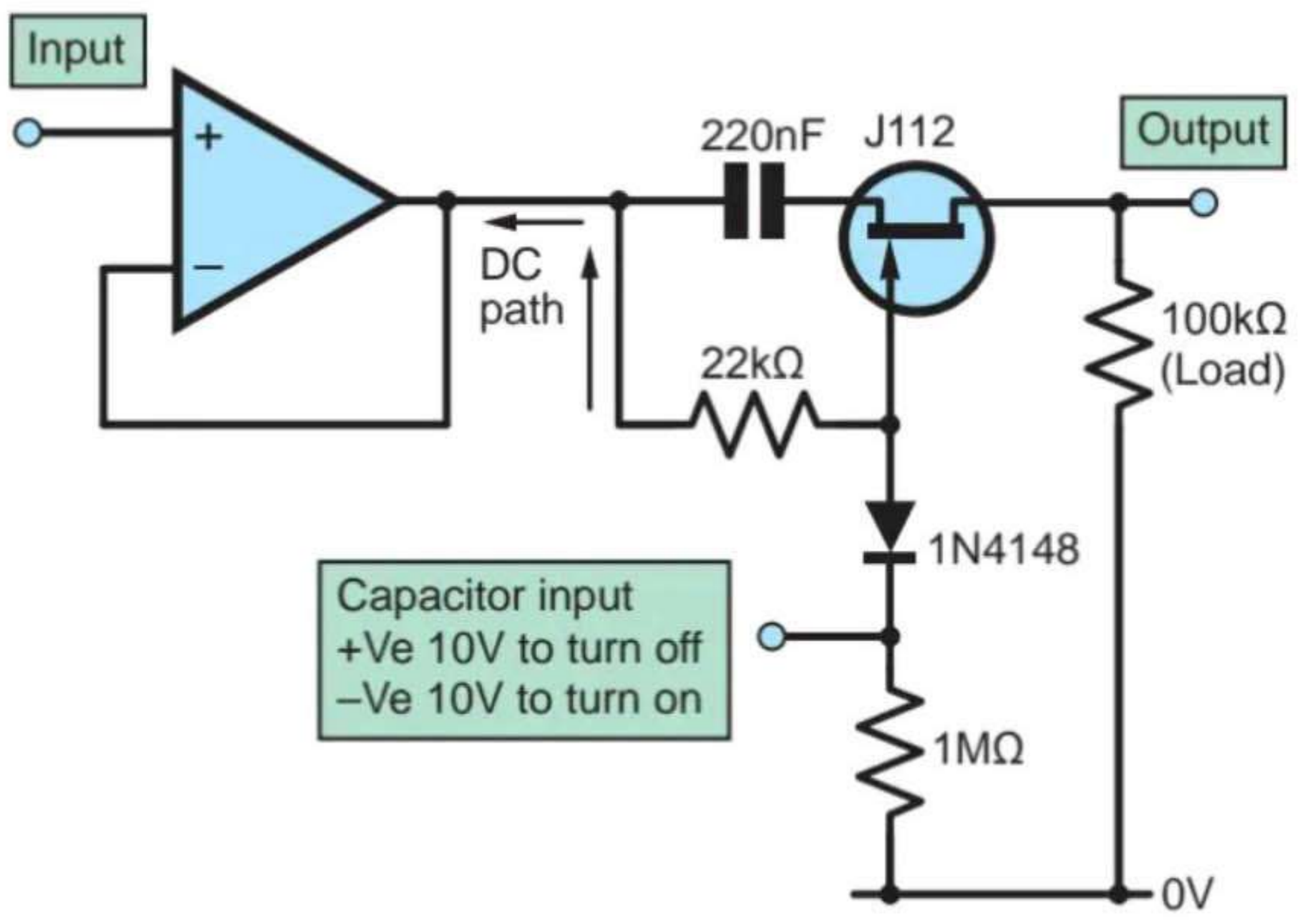


Fig.83: the AC-coupling arrangement for a JFET with bootstrapped DC path.

I built one section of this circuit (Fig.84[b]) and found that the op amp clipped on the negative cycle first, but this was only in mute when the negative control voltage was added on. So it was not a problem. Making a discrete version of this discontinued IC warrants further research, since the distortion was 0.0015% at $1V_{RMS}$ (0.01% at $4V_{RMS}$) into a high impedance load.

However, it was a bit worse than the diode bootstrapping method in my initial experiments.

Interestingly, the JFETs in the 2402 are P-channel types. This results from a similar ion implantation process to that used for Texas Instrument's Bi-Fet TL0xx series op amps. They also have P-channel JFETs, which is the outcome when made in association with the standard NPN bipolar transistor IC process.

Switch block

A switch block I developed based on virtual-earth topology is shown in Fig.85.

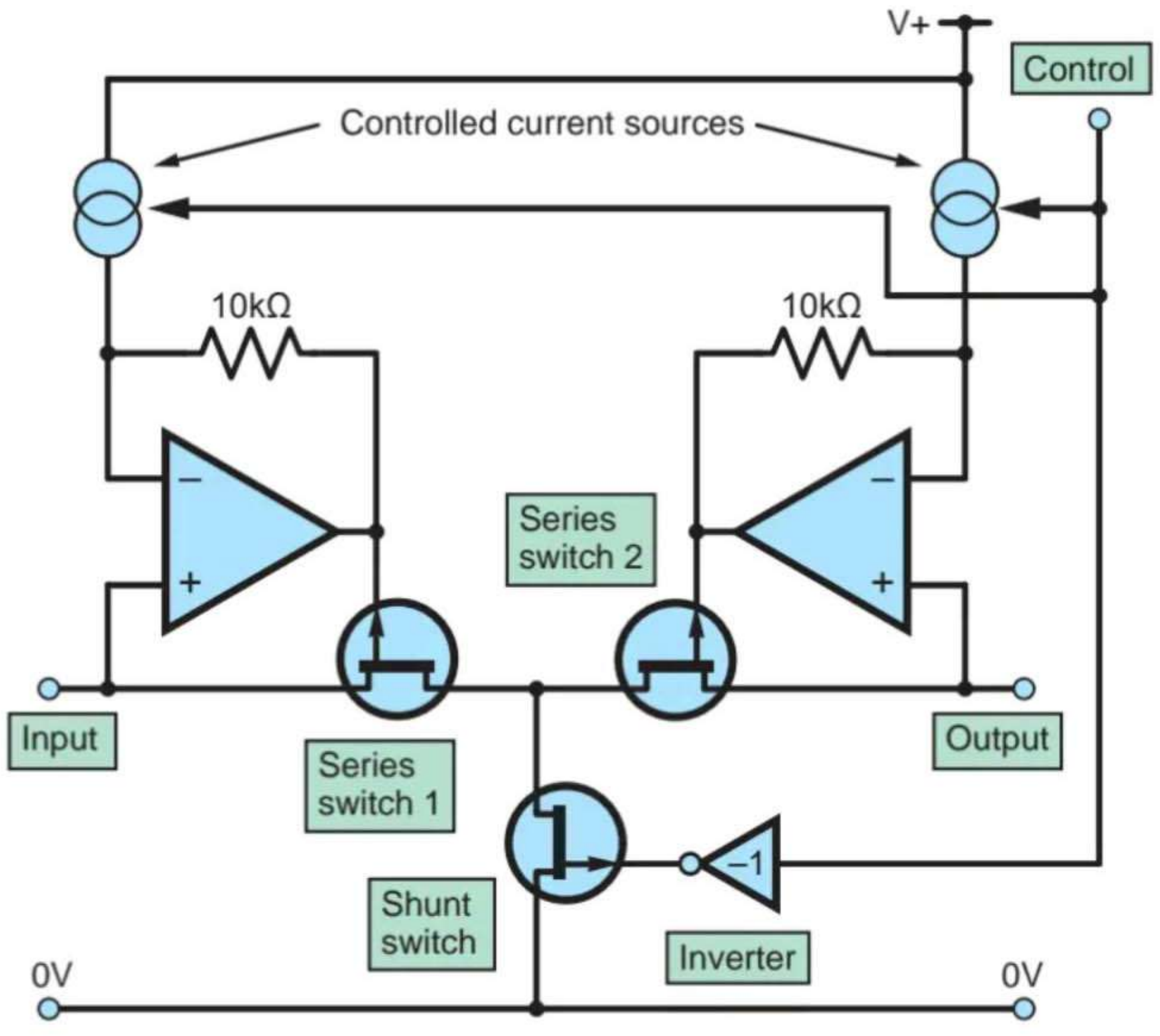


Fig.84(a): the SSM2402 chip uses interesting op amp bootstrapping with three FETs in a bidirectional series/shunt/series network.

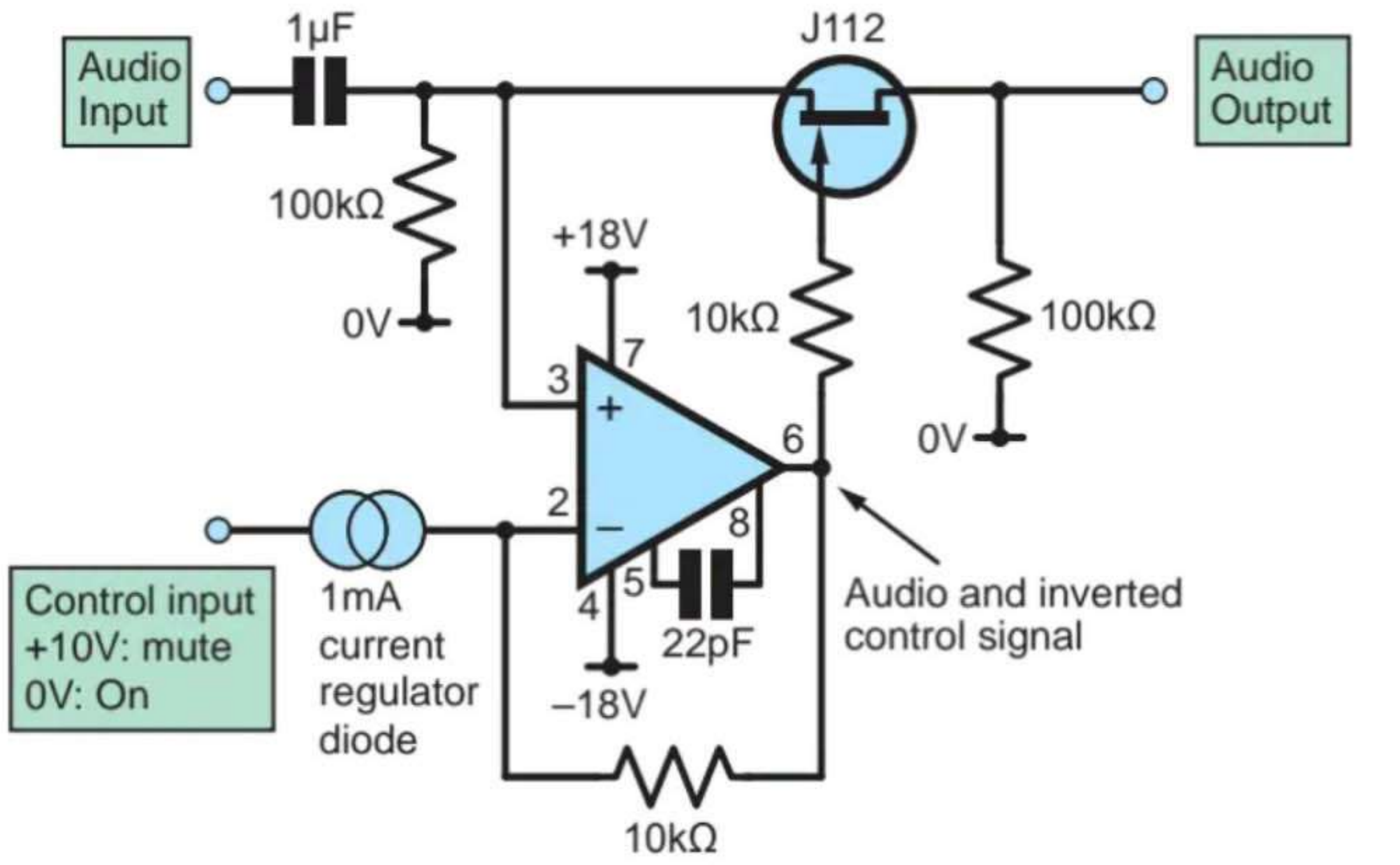


Fig.84(b): I found this discrete version of the SSM2402 concept to work OK.

This system is really the minimum required for a decent audio switch. It avoids the need for bootstrapping because the signal voltage across switch TR1 is very low when it is conducting, so V_{GS} is also very low.

A shunt JFET switch (TR2) attenuates the input in conjunction with R1 when the switch is off, preventing break-over. Noise is slightly higher with the inverting op amp than with the voltage-follower op amp system (a unity-gain inverting op amp has a noise gain of two). A good compromise between noise, input impedance and distortion is to use 4.7kΩ for R1 and R2.

My circuit uses blocking capacitors C1 and C2 because FET input op amps, especially the low-noise ones, have become expensive. The two capacitors are connected back-to-back in the series chain, effectively making a composite non-polarised capacitor, which reduces their distortion, especially with tantalum types.

I used a P-channel J175 for TR2 with a slight reduction in maximum attenuation. An inverter stage could be added to allow two N-channel J112 FETs to be employed (as shown in Fig.84[b]). C5 is a phase-lead compensation capacitor to compensate for the capacitance of TR2 and layout capacitance around pin 2. The measured distortion was 0.001% at 10V RMS from 20Hz to 20kHz.

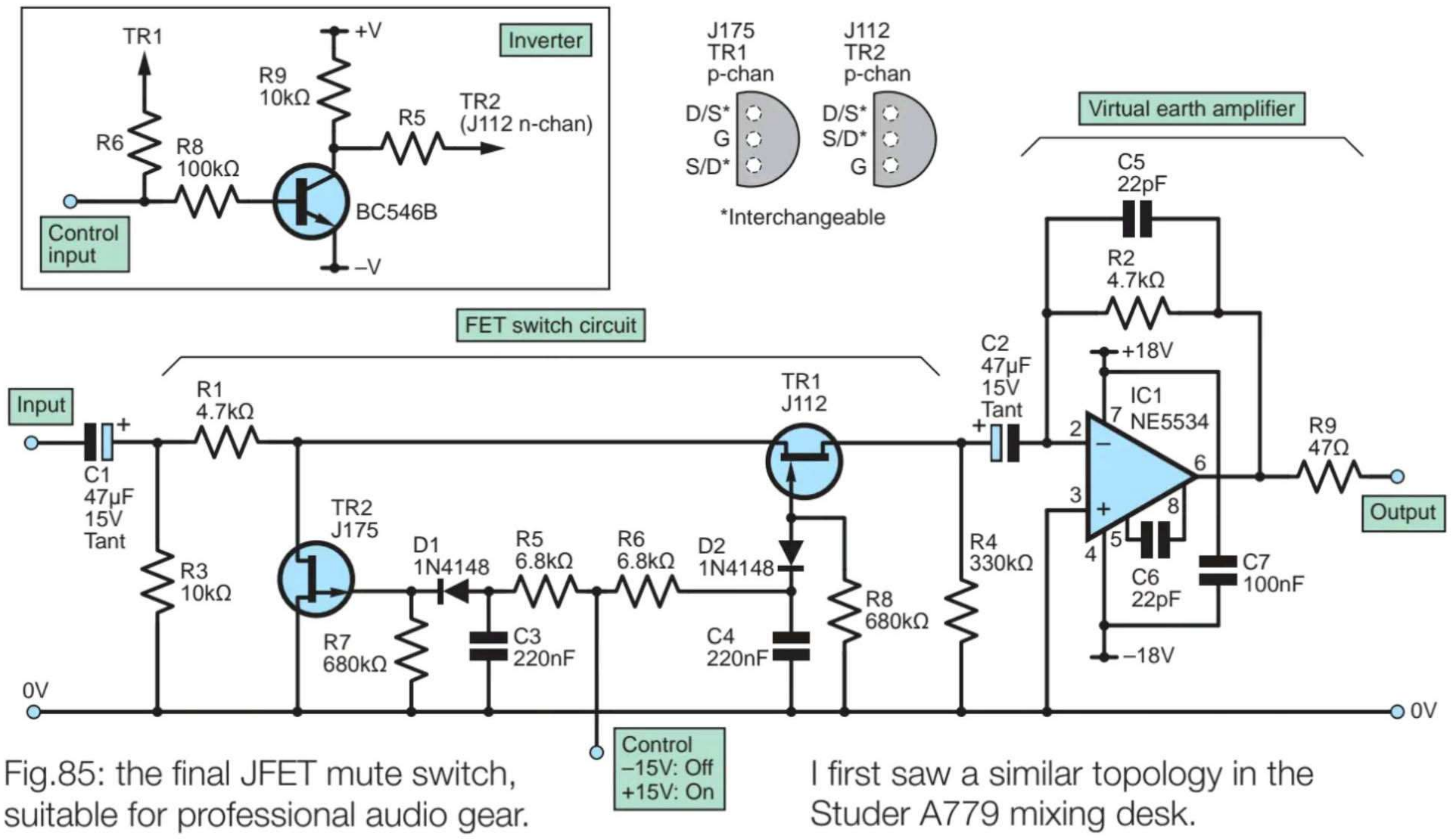

Inverter
TR1
R9
10kΩ
+V
R6
R8
100kΩ
R5
TR2
(J112 n-chan)
BC546B
–V
Control input
J175
TR1
p-chan
J112
TR2
p-chan
D/S*
G
S/D*
*Interchangeable
Virtual earth amplifier
FET switch circuit
Input
R1
4.7kΩ
C1
47µF
15V
Tant
R3
10kΩ
TR2
J175
R7
680kΩ
D1
1N4148
R5
6.8kΩ
C3
220nF
R6
6.8kΩ
D2
1N4148
C4
220nF
TR1
J112
R8
680kΩ
C2
47µF
15V
Tant
R4
330kΩ
C5
22pF
R2
4.7kΩ
+18V
IC1
NE5534
C6
22pF
C7
100nF
–18V
R9
47Ω
Output
0V
Control
–15V: Off
+15V: On

Fig.85: the final JFET mute switch, suitable for professional audio gear.

I first saw a similar topology in the Studer A779 mixing desk.

Attenuation was about -90dB at 10kHz with the J175. Ultimately, an electronic switch needs around 16 components just to approach the qualities of a simple mechanical switch.

Ramping

Each gate has two RC networks, R5/C3 and R6/C4, to ramp the JFET gate voltages, giving almost a fade rather than an abrupt transition. It is a distorted fade when switching a high-amplitude sinewave, but it is less noticeable than a click when music is switched.

This is done on the SSM2402 IC by special on-chip silicon nitride capacitors, ensuring smooth break-before-make action. There was a 2412 chip variant with faster switching for broadcast use.

More circuit tricks

I have seen some tricks applied on the Soundcraft 6000 series desks that are worth looking at. It is possible to have a DC-blocking capacitor in series with the lower JFET switch, as shown in Fig.86. This capacitor is thus out of the audio path when the whole switch circuit is passing audio.

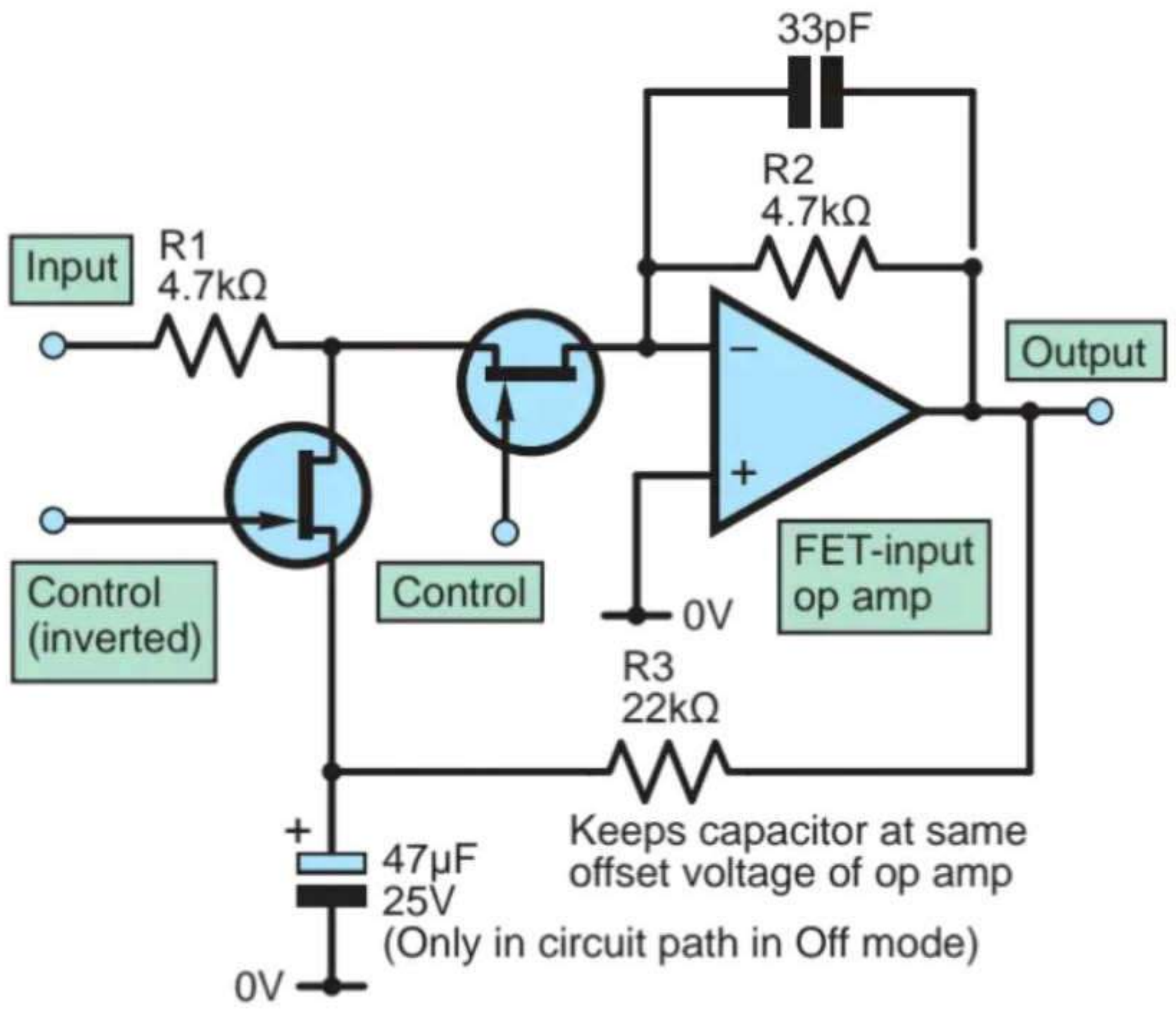

Input
R1
4.7kΩ
Control
(inverted)
Control
33pF
R2
4.7kΩ
Output
FET-input
op amp
0V
R3
22kΩ
47µF
25V
0V
Keeps capacitor at same offset voltage of op amp
(Only in circuit path in Off mode)

Fig.86: you can put the DC blocking capacitor in series with the lower JFET instead of the op amp input.

Resistor R3 is used to equalise the DC offset levels, preventing clicks. This technique needs FET-input op amps, such as the expensive OPA1641, since the bias currents from bipolar devices like my favourite, the NE5534, will cause clicks.

Minimising R_{ON}

Although N-channel JFETs are on at 0V gate bias, a bit of extra positive bias (200-300mV) can minimise R_{ON} and consequently distortion. For a typical J112, at 0V V_{GS}, R_{ON} = 38Ω, but with a V_{GS} of 300mV, it drops to 28Ω.

This can be achieved by the potential divider action of a 10MΩ resistor across the diode and a 680kΩ resistor to ground from the gate with a +7.5V control signal. I only got 300mV while there was no signal passing through, since the 'diode action' of the gate generated a small negative voltage, so I did not think this technique was worthwhile.

It's also a problem at high temperatures due to the gate's diode leakage being temperature-dependent.

A changeover switch

By combining two JFET switch circuit blocks (Fig.85) with a shared virtual earth op amp circuit, a changeover switch can be made, as shown in Fig.87. This can be driven by the flip-flop switch circuit from Fig.59. Since its control outputs are complementary -V and +V, all the JFETs can be N-channel J112s or even J111s because the negative voltage can go up to -16V.

Interlocking switches

The Quad selection circuit is the simplest electronic channel selector I've seen and makes the circuit given in Fig.59, with its debouncing, seem over the top. This circuit (Fig.88) is ideal for use with CD4066 devices.

Contact bounce does not seem to be a serious problem in practice with Hi-Fi input selection. The occasional 'jump' is not as catastrophic in audio as it would be in an industrial control system. Sufficient debouncing can usually be achieved with capacitors across switches, especially with slow 4000-series logic.

The Quad circuit has a capacitor wired across S2 to ensure the circuit always defaults to this position when powered. It can be moved to whatever position is required at power-on.

40xx series logic gates alone often had insufficient output current for high brightness with older technology LEDs, which is why the Quad circuit had driver transistors. The latest high-efficiency LEDs are bright enough at 1mA, such as the red TruOpto (Rapid 55-2190) or Kingbright (72-8989). So, if one of them is used, the driver

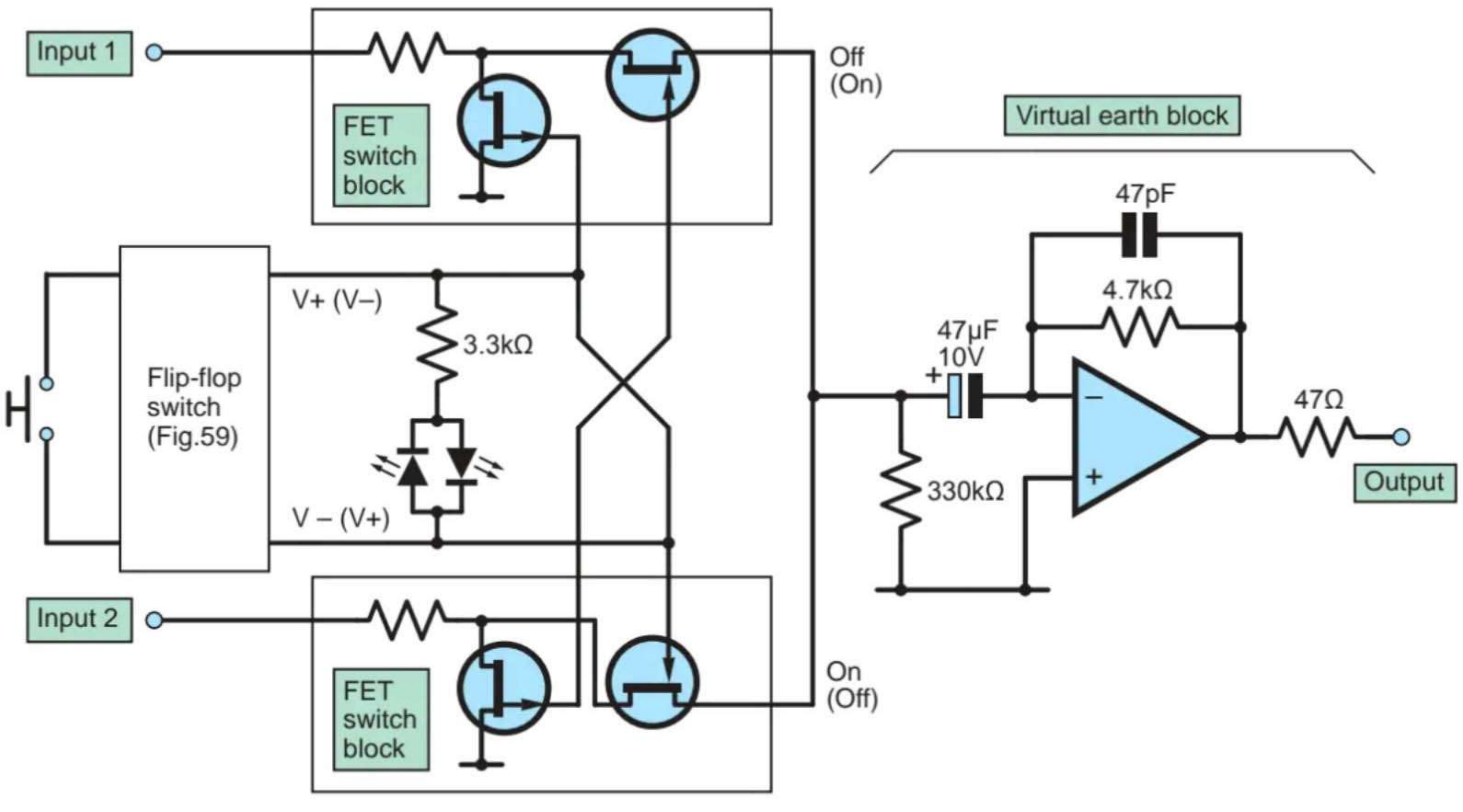

Input 1
FET switch block
Off
(On)
Flip-flop switch
(Fig.59)
V+ (V–)
3.3kΩ
V – (V+)
Input 2
FET switch block
On
(Off)
Virtual earth block
47pF
4.7kΩ
47µF
10V
330kΩ
47Ω
Output

Fig.87: a changeover switch can be made by feeding two mute circuits into one op amp. The op amp flip-flop (Figs.59 & 80) can control this. All FETs can be J112s.

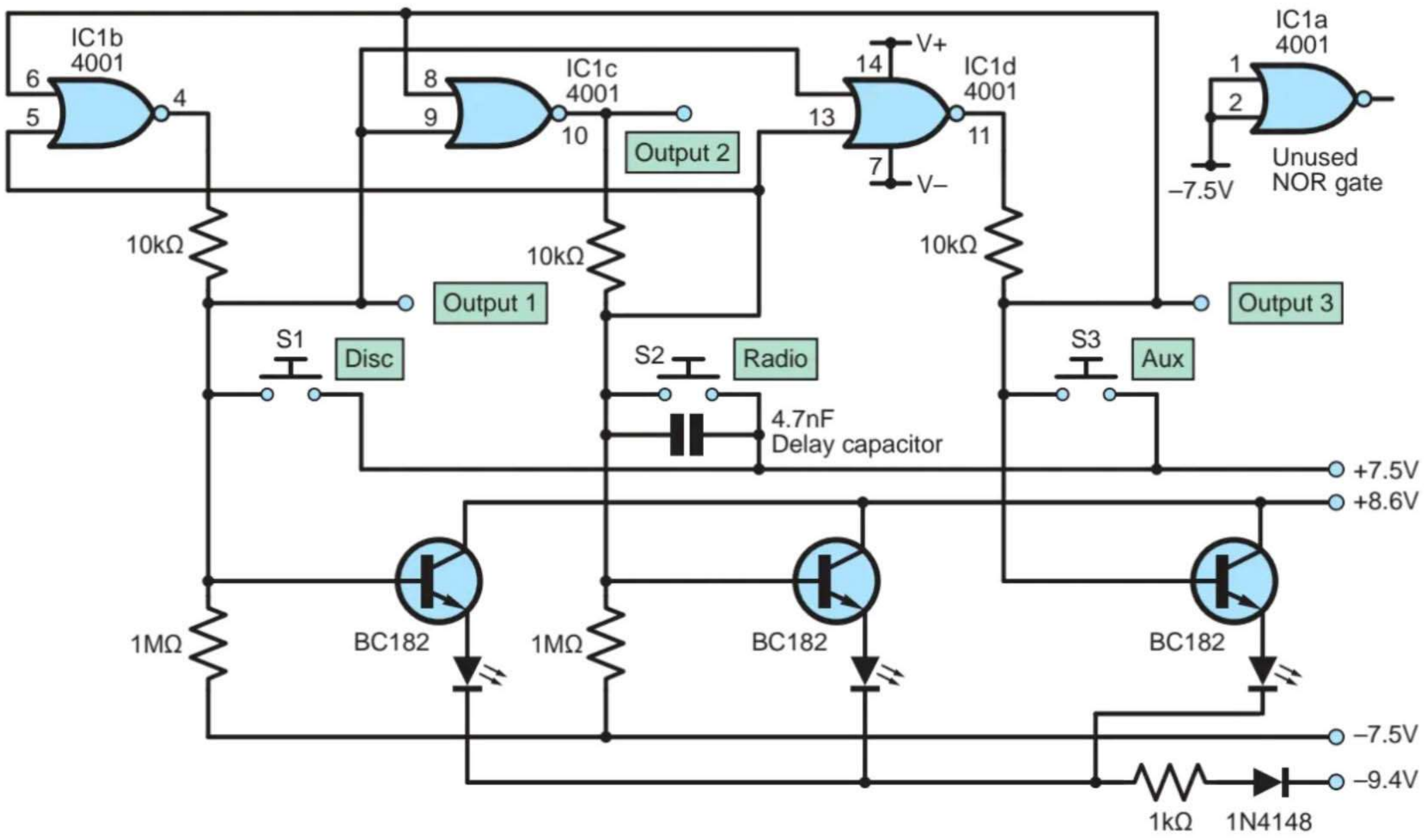

IC1b
4001
6
5
4
8
9
IC1c
4001
10
Output 2
14
V+
13
IC1d
4001
11
7
V–
IC1a
4001
1
2
–7.5V
Unused
NOR gate
10kΩ
10kΩ
10kΩ
Output 1
Output 3
S1
Disc
S2
Radio
S3
Aux
4.7nF
Delay capacitor
+7.5V
+8.6V
1MΩ
BC182
1MΩ
BC182
BC182
–7.5V
–9.4V
1kΩ
1N4148

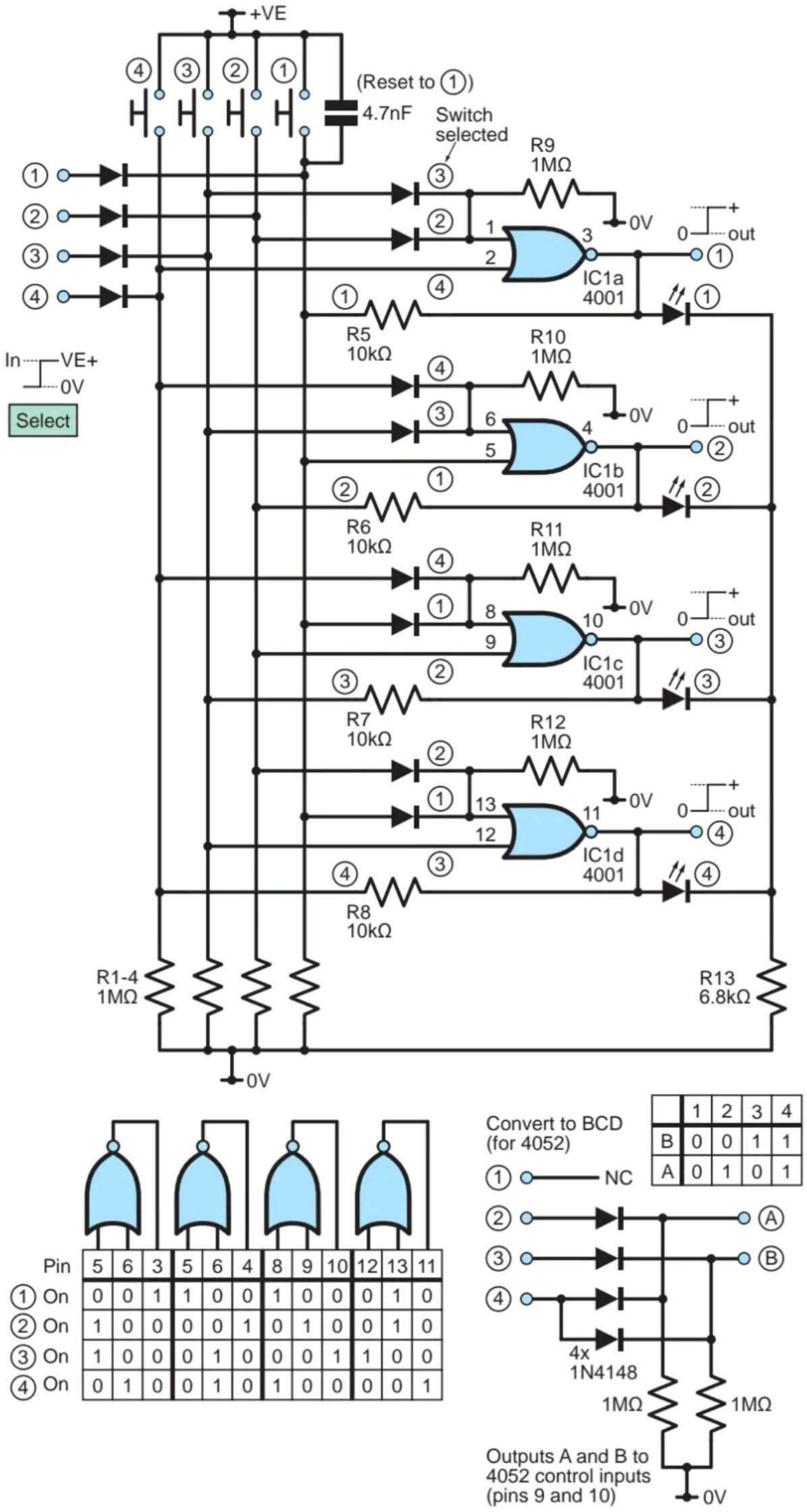
Fig.88: the Quad 34 3-channel selector switch circuit. The power rails for the ICs are +7.5V (pin 14) and -7.5V (pin 7), so the audio signal can be DC-biased at 0V.

transistors can be dispensed with, as shown in Fig.89.

The circuit only consumes 0.9mA at 9V and 1.9mA at 15V.

The Quad circuit had only three positions, but the package has a spare NOR gate. I had it redesigned by Grant Stevens to make it a four-position switch with the addition of some diodes, as shown in Fig.90. This ties up nicely with the 4066 chip that has four switches. The start-up position capacitor has been increased in value to allow for the slow ramp-up of some power supplies.

Note that when feeding a high-impedance CMOS gate with diodes, it is often necessary to follow the diode with a 100kΩ to 1MΩ pull-down resistor (R9 to R12 in Fig.90) due to the leakage current. Grant also designed a decimal to binary-coded decimal (BCD) diode logic circuit to drive a 4052 audio switch, described below.


+VE
④
③
②
①
(Reset to ①)
4.7nF
Switch
selected
R9
1MΩ
0V
①
②
③
④
In
VE+
0V
Select
③
②
1
2
3
IC1a
4001
0
+
out
①
①
④
R5
10kΩ
R10
1MΩ
④
③
6
5
4
IC1b
4001
②
②
①
R6
10kΩ
R11
1MΩ
④
①
8
9
10
IC1c
4001
③
③
②
R7
10kΩ
R12
1MΩ
②
①
13
12
11
IC1d
4001
④
④
③
R8
10kΩ
R1-4
1MΩ
R13
6.8kΩ
0V
Convert to BCD
(for 4052)
NC
Ⓐ
Ⓑ
4x
1N4148
1MΩ
1MΩ
Outputs A and B to
4052 control inputs
(pins 9 and 10)
0V

Pin	5	6	3	5	6	4	8	9	10	12	13	11
① On	0	0	1	1	0	0	1	0	0	0	1	0
② On	1	0	0	0	0	1	0	1	0	0	1	0
③ On	1	0	0	0	1	0	0	0	1	1	0	0
④ On	0	1	0	0	1	0	1	0	0	0	0	1

	1	2	3	4
B	0	0	1	1
A	0	1	0	1

Fig.90: why waste a gate in a quad package? Here is the Quad circuit expanded to four buttons. The inset diagram is a diode logic driver for use with a 4052.

The Playmaster Series 200

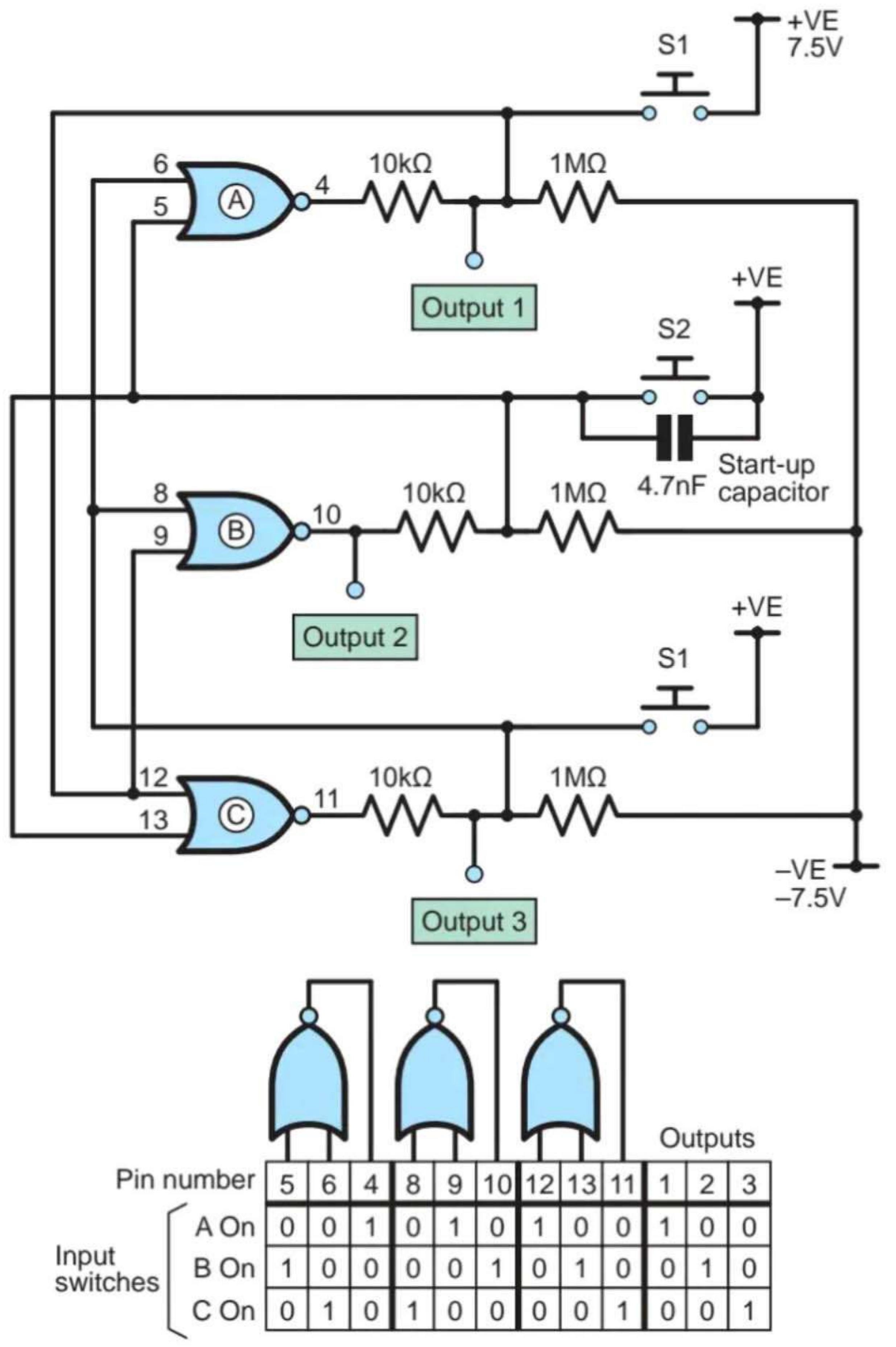
One of the first DIY amplifiers using a similar switching system was the Playmaster Series 200, published in *Electronics Australia* in March 1985 (page 38). It used 4052s (two four-way switches in a package) and 4011 NAND gate flip-flops, as shown in Fig.91. Some may prefer this circuit because the switches go to ground.

Using a couple of diodes, a backup voltage source could be added to the power pins of the CMOS chips to memorise the switch positions. The current consumption of these devices is below 10µA; I used a 100µF 10V low-leakage tantalum capacitor, which gave a backup time of about half an hour. Of course, the LEDs have to be connected to the anode side of diode D9.

The 4052 is run from split power rails to accommodate the bipolar audio. The control signal can be normal logic levels of 0V and +5V because there is internal-level shifting circuitry in the 4052. Breakover occurs at 16V peak-to-peak and clipping at 13V peak-to-peak.

The original circuit used a BCD decoder made from another 4011 to drive the LEDs shown in Fig.92. A single 4052 was originally used, switching both left and right channels. The original op amp was a TL071, but this has a phase inversion at negative clipping when used as a follower, so it was changed to an NE5534.

Attenuation was only -40dB at 10kHz on the messy breadboard shown in Fig.93, but this soon cleared up when the unused inputs were grounded. If the inputs are all buffered, which they should be to protect the delicate CMOS gates, it is around -90dB. The distortion was surprisingly low and flat with frequency: 0.0006% at 1V RMS, rising to 0.004% at 4V RMS.


S1
+VE
7.5V
6
5
Ⓐ
4
10kΩ
1MΩ
Output 1
+VE
S2
8
9
Ⓑ
10
10kΩ
1MΩ
4.7nF
Start-up
capacitor
Output 2
+VE
S1
12
13
Ⓒ
11
10kΩ
1MΩ
–VE
–7.5V
Output 3
Outputs
Input
switches

Pin number	5	6	4	8	9	10	12	13	11	1	2	3
A On	0	0	1	0	1	0	1	0	0	1	0	0
B On	1	0	0	0	0	1	0	1	0	0	1	0
C On	0	1	0	1	0	0	0	0	1	0	0	1

Fig.89: a simplified Quad circuit with logic analysis by Grant Stevens.

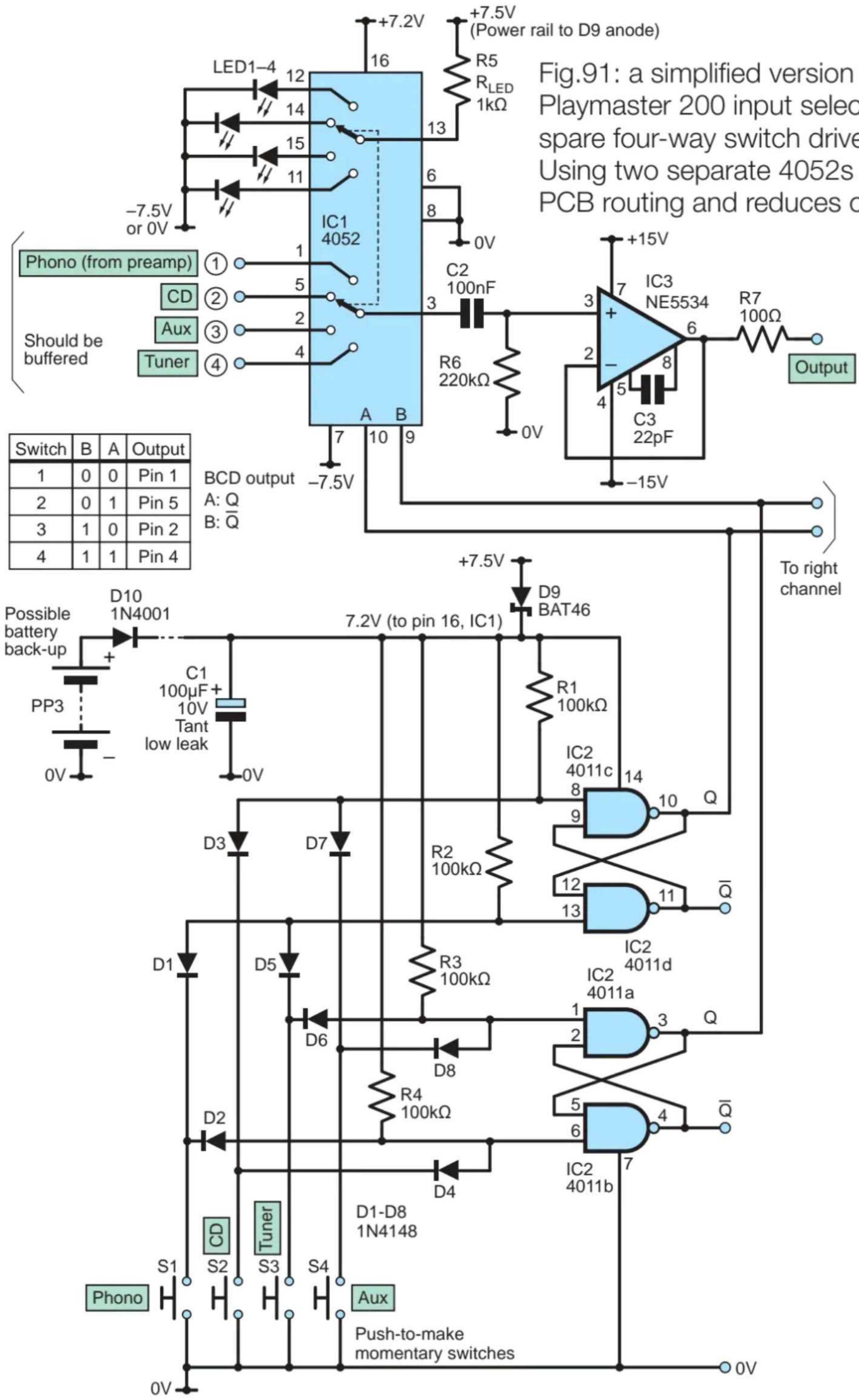


Fig.91: a simplified version of the Playmaster 200 input selector; the spare four-way switch drives the LEDs. Using two separate 4052s simplifies PCB routing and reduces crosstalk.

Switch	B	A	Output
1	0	0	Pin 1
2	0	1	Pin 5
3	1	0	Pin 2
4	1	1	Pin 4

+7.2V
+7.5V
(Power rail to D9 anode)
R5
R_{LED}
1kΩ
LED1–4
−7.5V
or 0V
IC1
4052
Phono (from preamp)
CD
Aux
Tuner
Should be buffered
C2
100nF
IC3
NE5534
+15V
−15V
R6
220kΩ
0V
C3
22pF
R7
100Ω
Output
BCD output
A: Q
B: Q̄
−7.5V
To right channel
+7.5V
D9
BAT46
D10
1N4001
Possible battery back-up
PP3
C1
100µF
10V
Tant
low leak
7.2V (to pin 16, IC1)
R1
100kΩ
IC2
4011c
R2
100kΩ
IC2
4011d
R3
100kΩ
IC2
4011a
R4
100kΩ
IC2
4011b
Q
Q̄
D1
D2
D3
D4
D5
D6
D7
D8
D1-D8
1N4148
S1
S2
S3
S4
Phono
CD
Tuner
Aux
Push-to-make momentary switches
0V

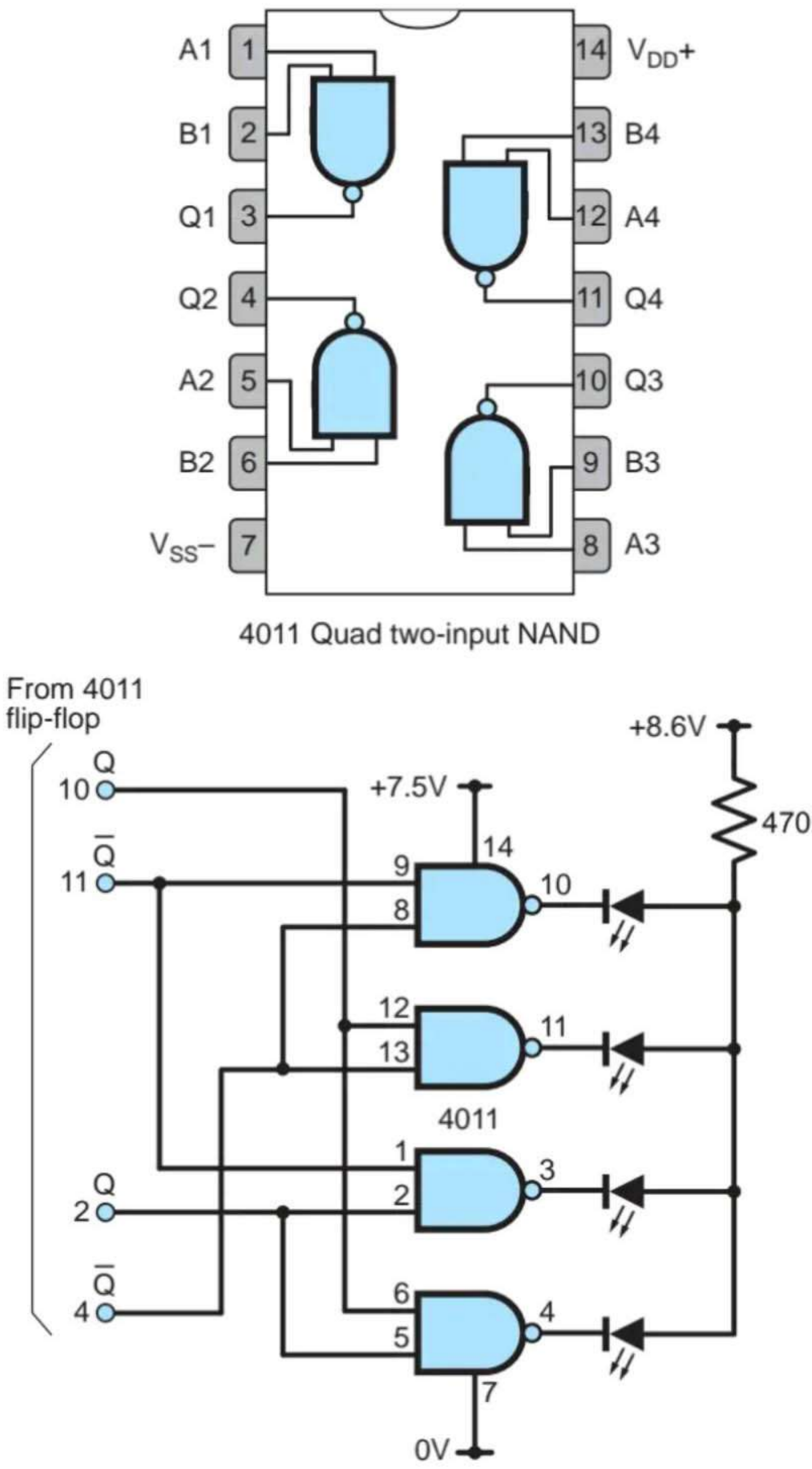


Fig.92: you can use this BCD decoder to drive the LEDs if needed. If the input is binary 11, LED 4 illuminates. When feeding in 00, LED 1 illuminates.

A1
B1
Q1
Q2
A2
B2
V_{SS}−
V_{DD}+
B4
A4
Q4
Q3
B3
A3
4011 Quad two-input NAND
From 4011 flip-flop
Q
Q̄
+7.5V
+8.6V
470Ω
4011
0V

Discrete button pushing

There is a mismatch between the high negative switching voltages of low-R_{ON} JFETs and standard logic chip families. There is also the hassle of providing dedicated logic power rails of different voltages. Solutions to these problems include using level-shifting circuits or designing one's own logic using discrete transistors.

Discrete logic is old-fashioned, but the requirements for audio switching are very simple. Many young engineers reach for an Arduino with scanned switches; contact bounce is important and can be dealt with in software.

There are not many discrete logic circuits about now, but I remember as a teenager (in the mid-1970s), there was a big thing about touch-switch channel selectors on TVs. I think some 1974 GEC C211X series sets used discrete transistors with little neon bulbs! As usual, I found a suitable circuit by perusing old *Wireless World* magazines.

In *WW* October 1974's Circuit Ideas section (page 380), I found an inspiring circuit by P. G. Hinch. I got it to work after a bit of semiconductor polarity flipping and resistor value changing; the final circuit is shown in Fig.94.

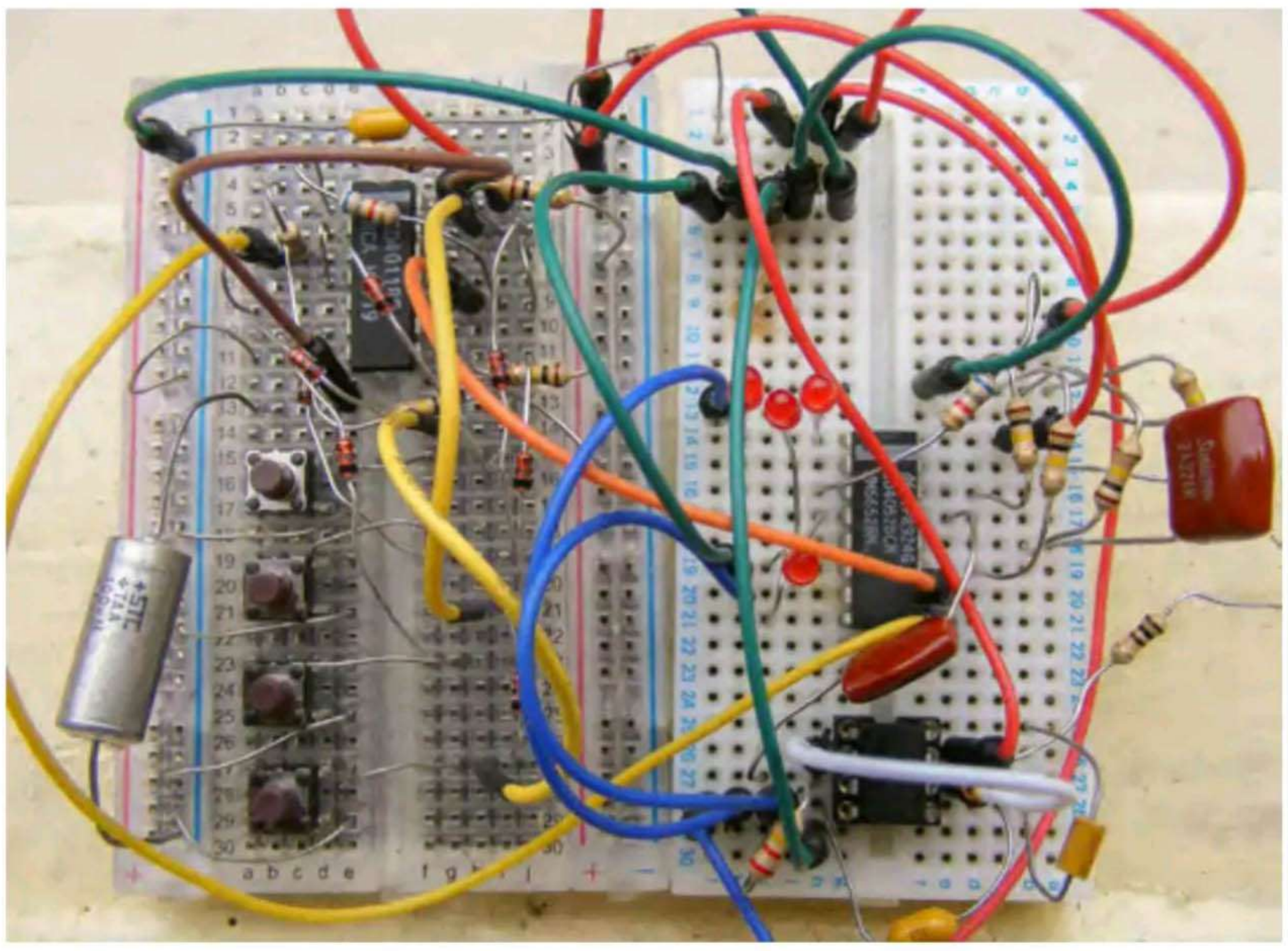


Fig.93: a Playmaster-style channel selector on a breadboard.

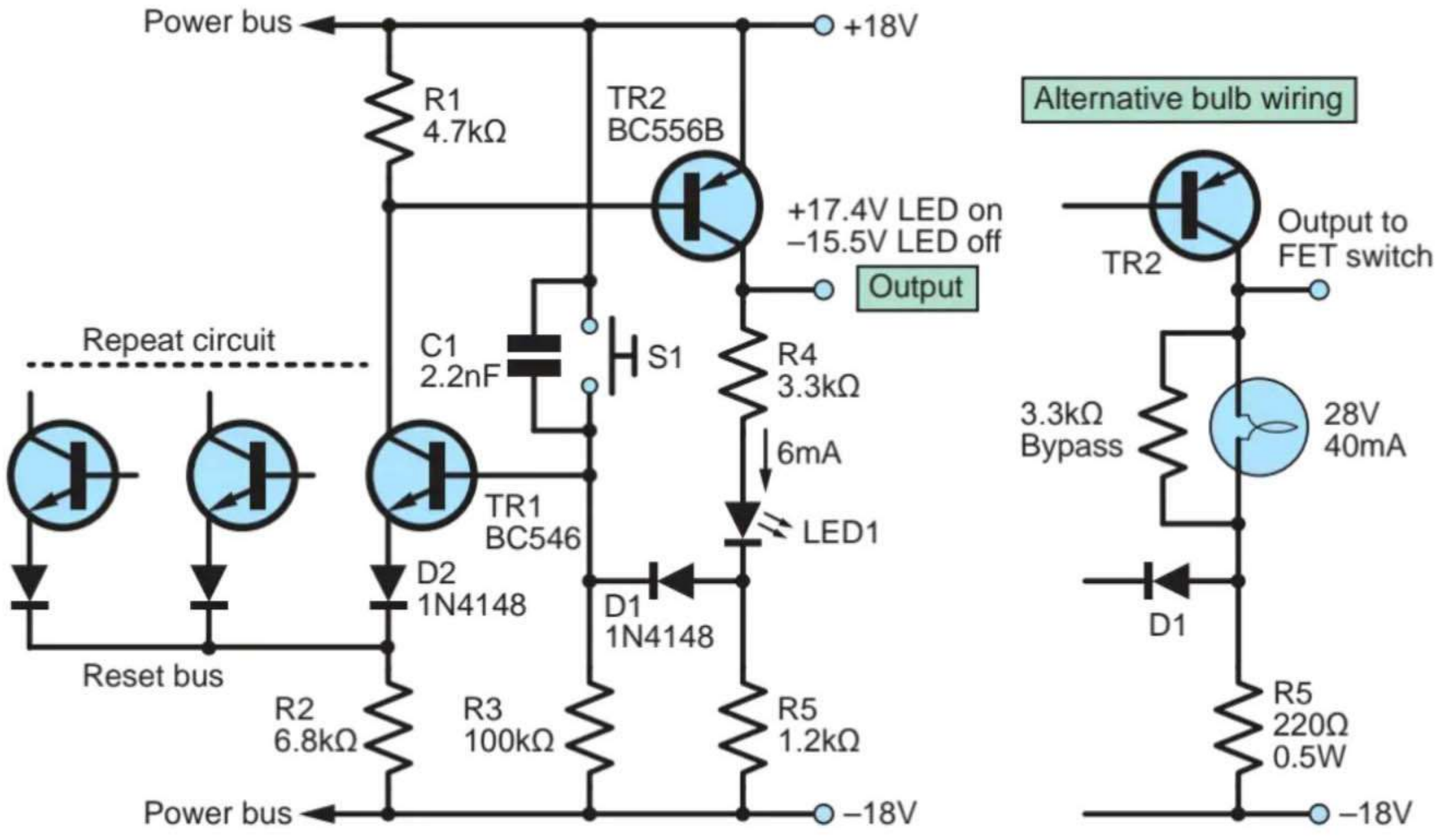


Fig.94: the final circuit (per switch) for the discrete interlocking circuit. The polarity is inverted from Hinch's design.

Power bus
+18V
R1
4.7kΩ
TR2
BC556B
+17.4V LED on
−15.5V LED off
Output
C1
2.2nF
S1
R4
3.3kΩ
6mA
LED1
Repeat circuit
TR1
BC546
D2
1N4148
D1
1N4148
Reset bus
R2
6.8kΩ
R3
100kΩ
R5
1.2kΩ
−18V
Alternative bulb wiring
TR2
Output to FET switch
3.3kΩ
Bypass
28V
40mA
D1
R5
220Ω
0.5W
−18V

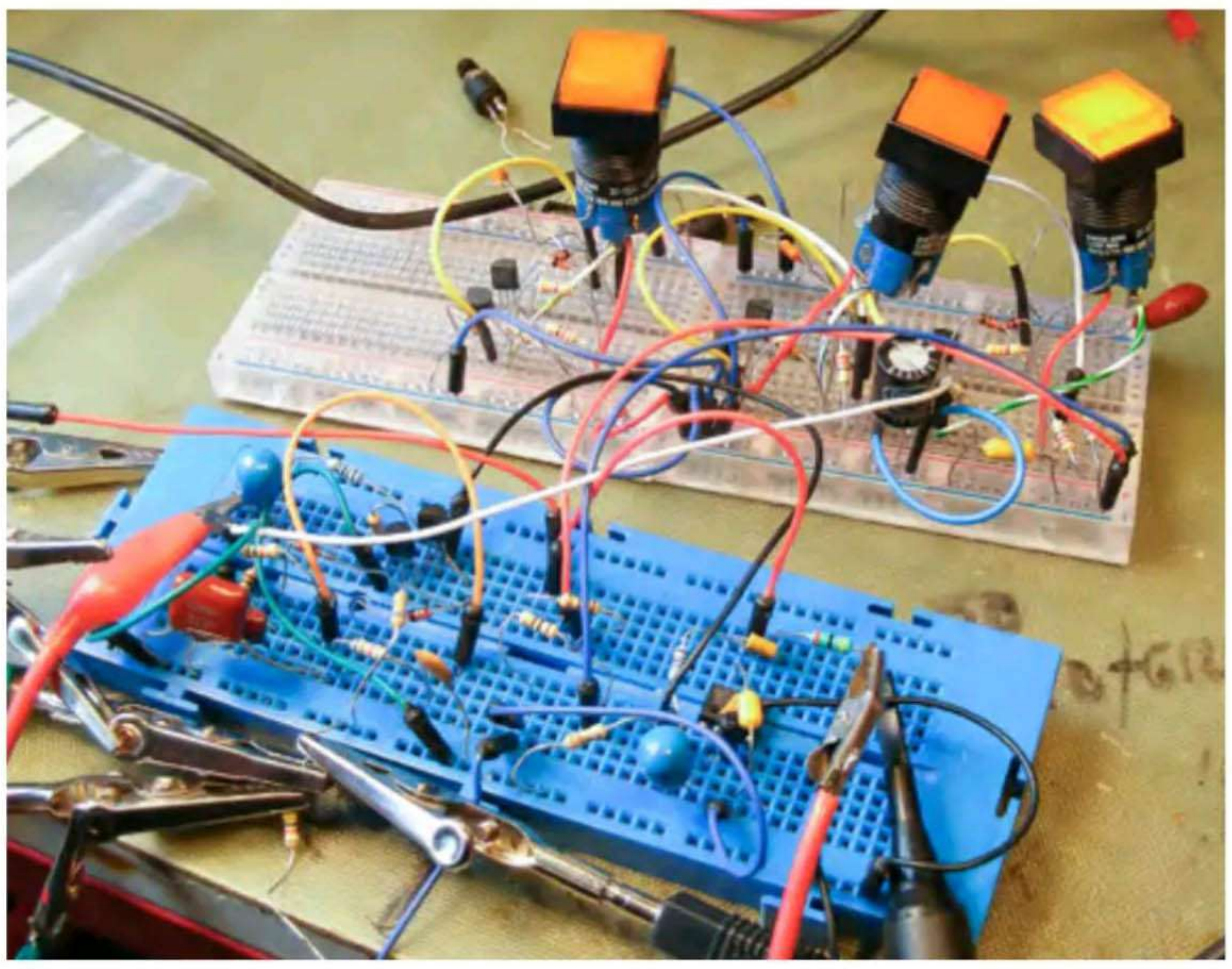


Fig.95: a discrete transistor interlocking switch circuit with filament light bulbs on a breadboard. The blue breadboard is the JFET audio switch from Fig.85.

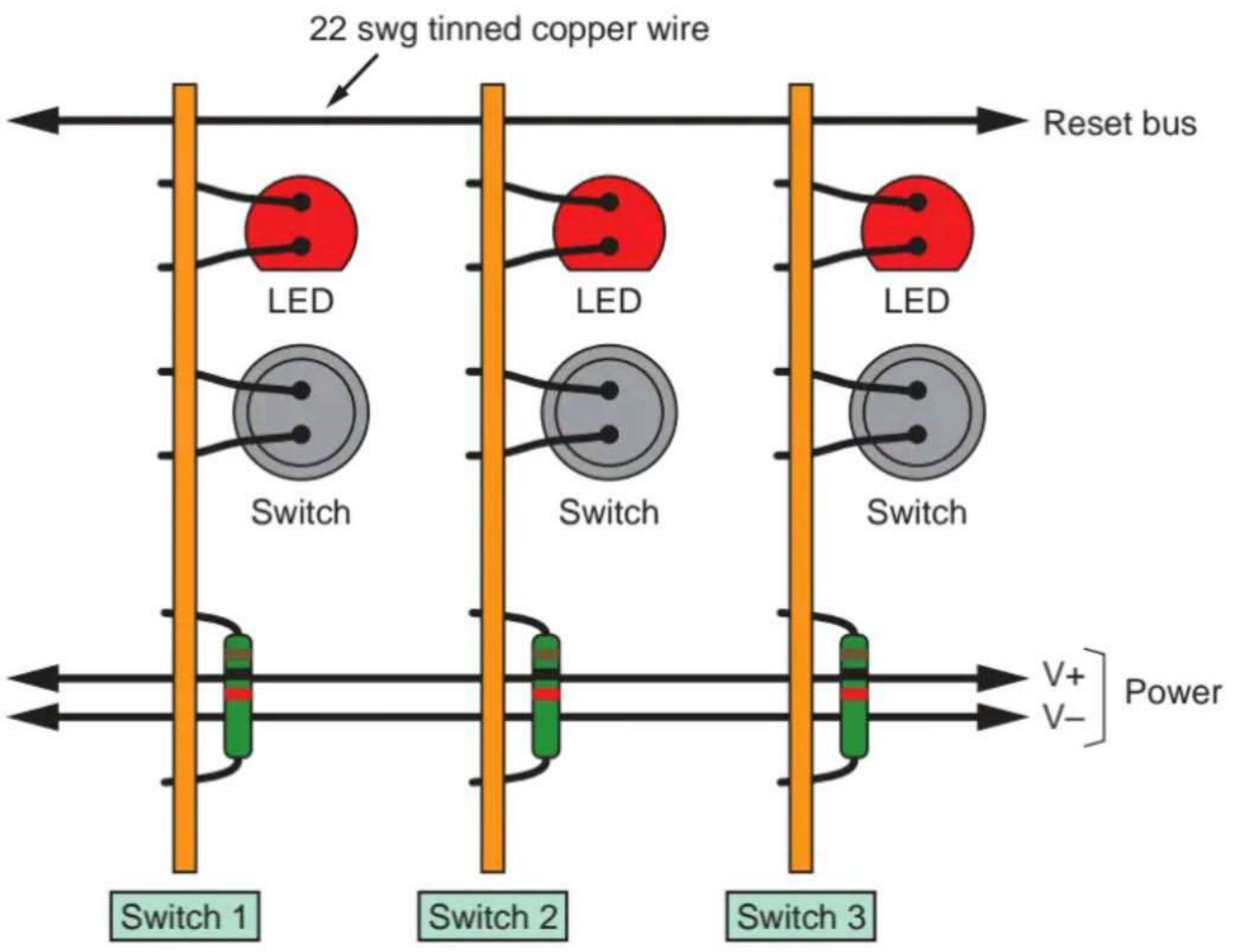

22 swg tinned copper wire
Reset bus
LED
LED
LED
Switch
Switch
Switch
V+
V–
Power
Switch 1
Switch 2
Switch 3

Fig.96: a suggested system for one PCB per switch, extendable as long as you like with busbars.

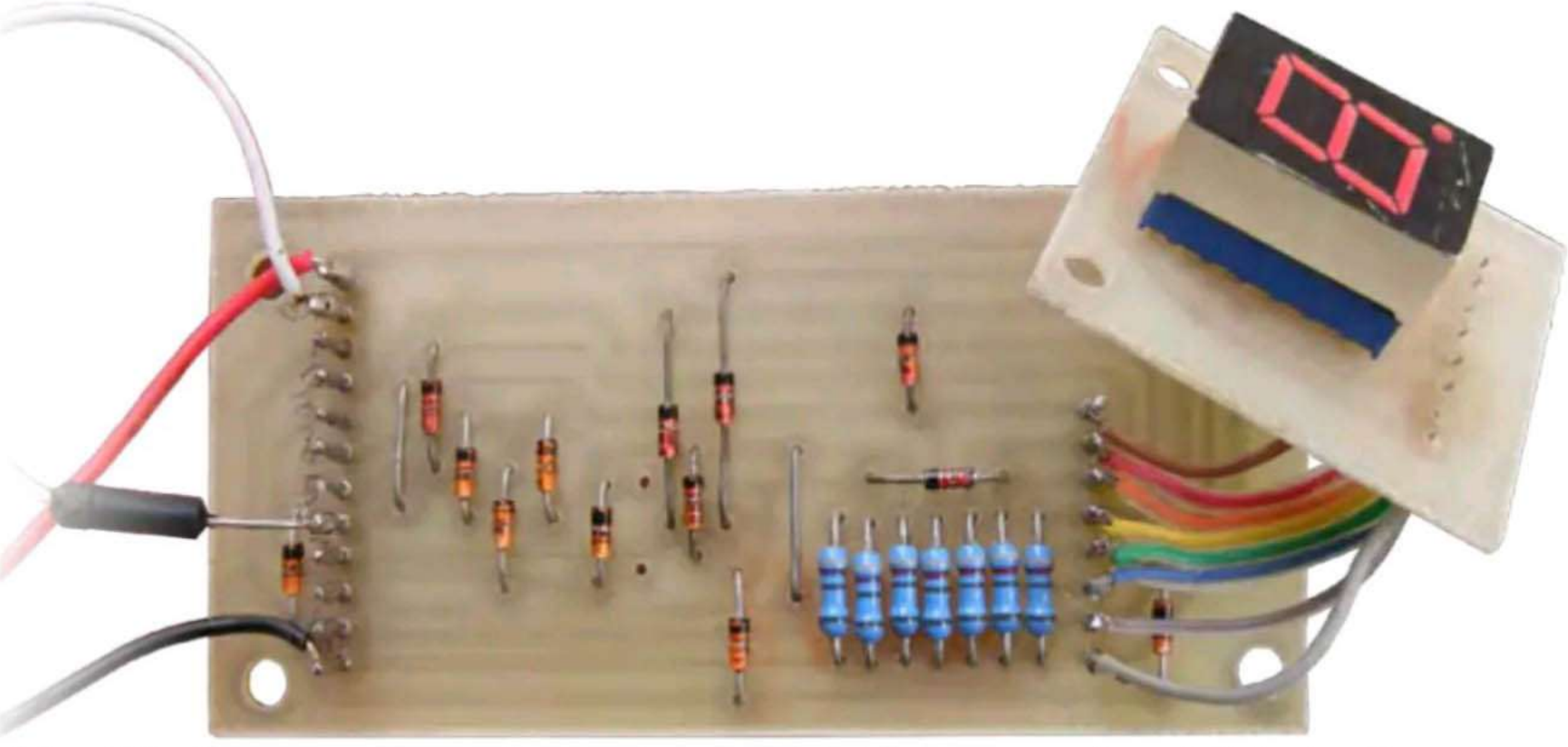


Fig.98: the rotary switch display PCB. I described this in the Modern Amateur Electronics Manual years ago (Supplement 24).

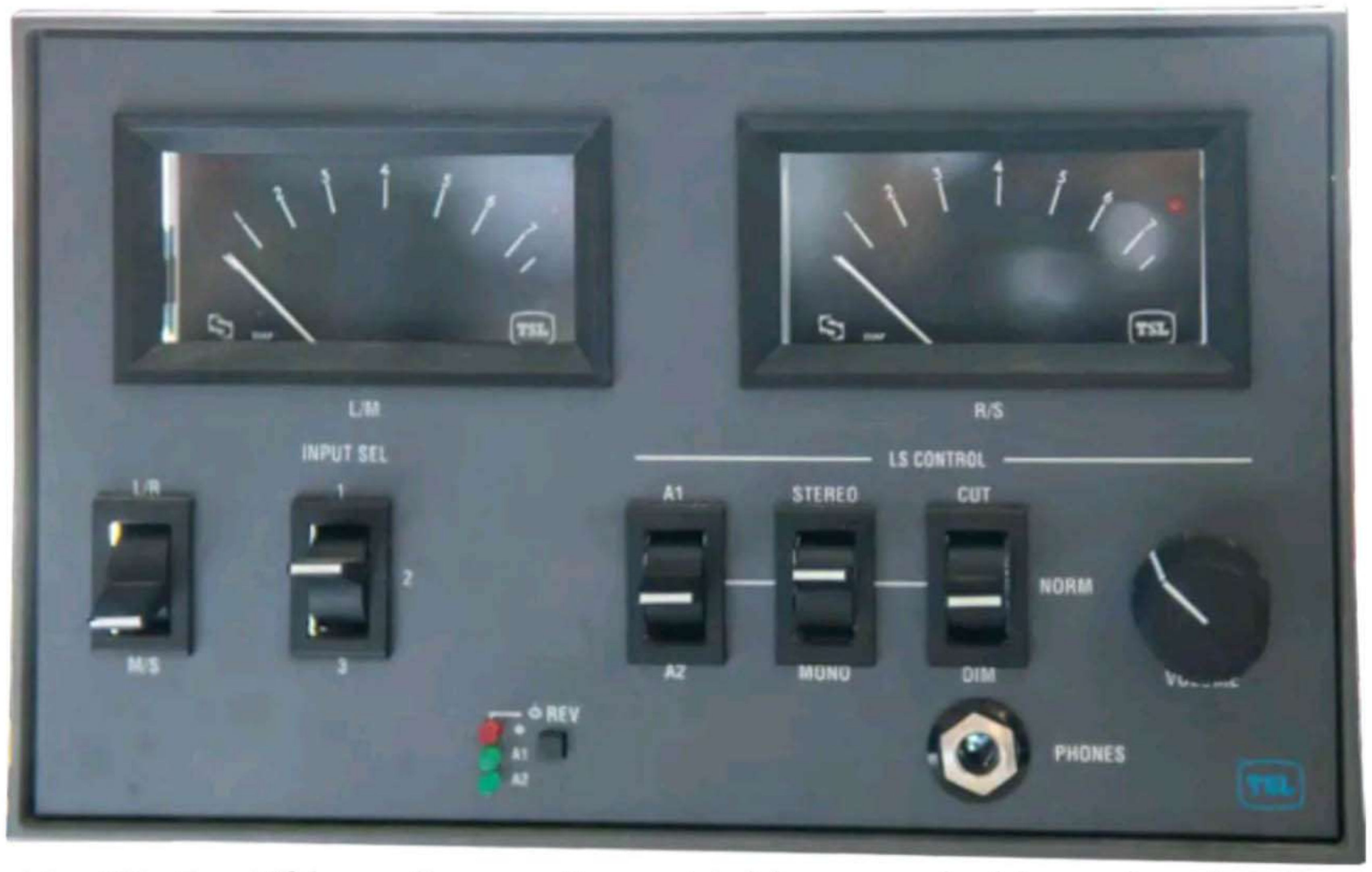

L/M
R/S
INPUT SEL
LS CONTROL
L/R
M/S
1
2
3
A1
A2
STEREO
MONO
CUT
DIM
NORM
REV
PHONES
TSL

Fig.99: the TSL audio monitor, a highly-regarded broadcast unit, excellent for driving LS3/5A speakers and monitoring the stereo compatibility of mixes.

The capacitor across the switch can be used to set which switch comes on when the circuit is powered. If you want S1 to be the default, increase C1 to 220nF.

This system is ideal for controlling the switch block in Fig.85, which needs +15V for on and -15V for off using standard op amp power rails.

Discrete transistor circuits also allow the LED current to be increased for higher brightness and can easily drive the 28V 40mA (eg, CML CM334) filament bulbs in EAO switches by changing R5 to 220Ω ½W. Unlike LEDs, bulbs do not require a series resistor (R4), so the bulb is wired from the collector of TR2 to R5.

If the bulb blows (which they do every 4000 hours), the interlocking ceases to function, so each bulb must have a 3.3kΩ resistor wired in parallel to ensure enough current flows to trigger the circuit.

The current consumption for the bulb circuit is 45mA from both rails. In this circuit, the bulbs are under-run at 23V, so they last a long time.

I had to prove the concept on a breadboard; see Fig.95. I've kept a whole bag of these momentary switches in stock for years – now I have a use for them!

The great thing about this discrete circuit is that it can be divided into a separate section for each switch. You can have a string of buttons as long as you like, as shown in Fig.96.

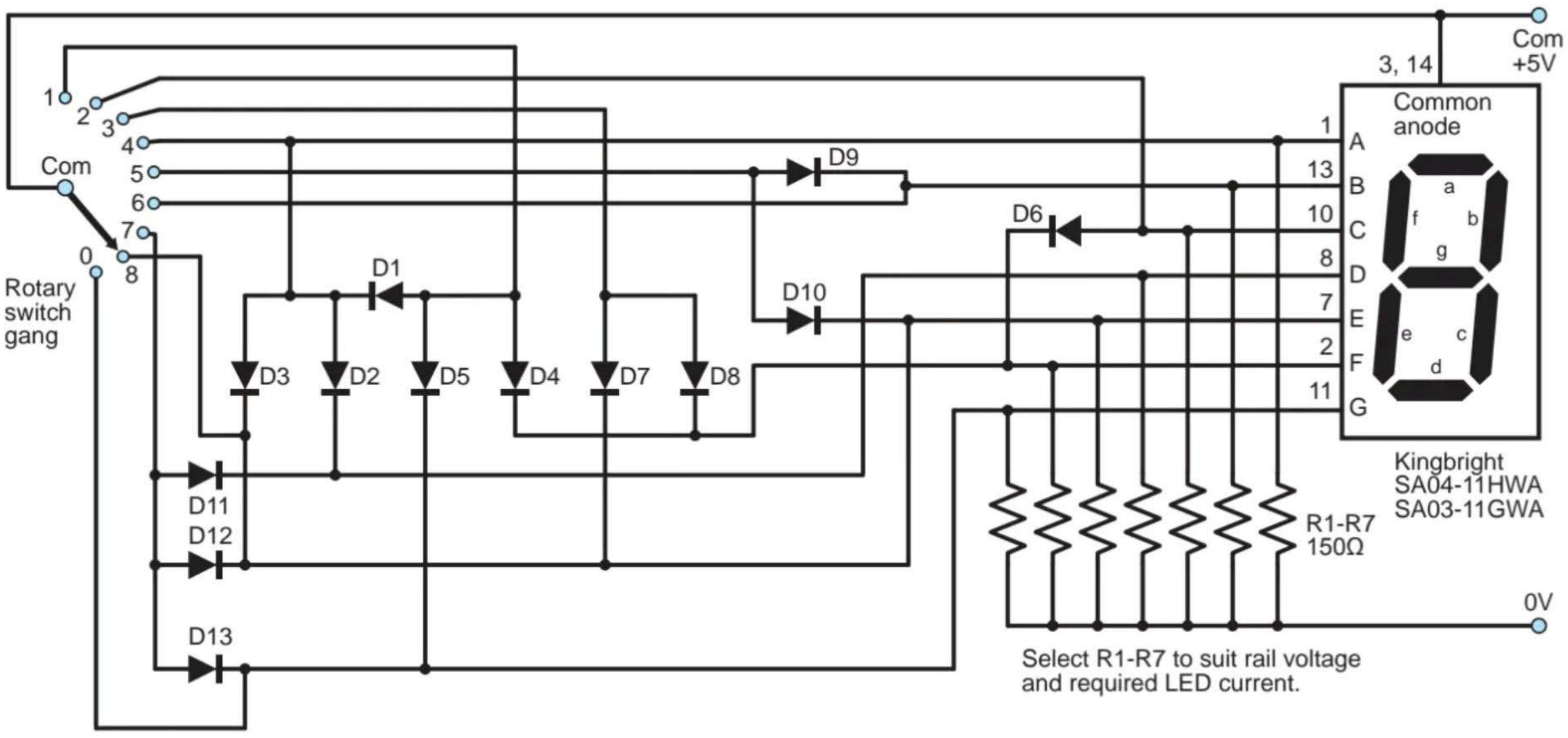

Com +5V
3, 14
Common anode
1
13
10
8
7
2
11
A
B
C
D
E
F
G
a
b
c
d
e
f
g
1
2
3
4
5
6
7
8
0
Com
Rotary switch gang
D1
D2
D3
D4
D5
D6
D7
D8
D9
D10
D11
D12
D13
Kingbright SA04-11HWA SA03-11GWA
R1-R7 150Ω
0V
Select R1-R7 to suit rail voltage and required LED current.

Fig.97: a diode-based seven-segment display driver for a rotary switch. You need an extra gang on the switch but avoid complex front-panel artwork.

Discrete diode circuits

Some switching logic is so simple it can be done with diodes, such as the rotary switch LED readout circuit shown in Fig.97.

There was an excellent story about this in the *Radio Constructor* series, In *Your Workshop*, May 1978. The assembled PCB is shown in Fig.98.

Doing it properly

The Television Systems Limited (TSL) analog audio broadcast monitor shown in Fig.99 exemplifies electronic switching and general construction. The whole audio signal path is dealt with in a self-contained plug-in Euro card at the back of the unit near the audio XLR connectors.

The interior is shown in Fig.100, the Euro card is shown in Fig.101 and the bilateral electronic switches used are Vishay's DG509.

The DG509 is a device that can operate from supply voltages of up to 44V maximum and it contains two four-way switches; a posh CD4052, in effect. The binary input to switch it is also the same. It costs around £4 from Mouser.

The front panel switches are all mechanically latching C&K paddle types, just controlling DC, which is fed to the Euro card by a grey ribbon connector.

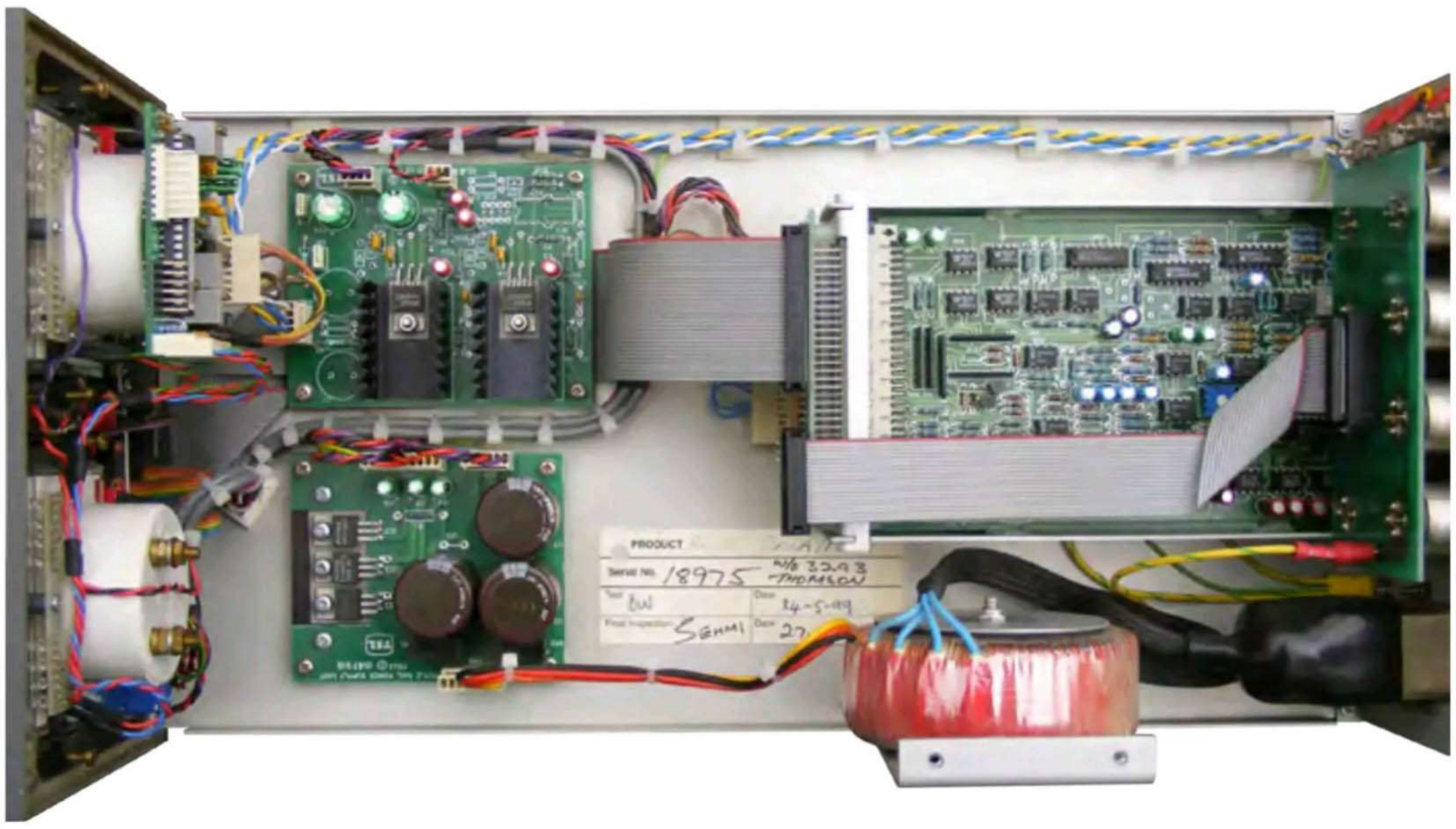


Fig.100: the interior of the TSL audio monitor, a great example of audio gear built properly. The audio processor card is in its own screened box.

New chips on the block

There are some new audio switch chips available. They are all surface-mounting low-voltage (5V) types, so they are difficult to incorporate in conventional ±15V powered op amp audio circuitry.

The Diodes Incorporated PI5A3157 has a 6Ω R_{ON} but it comes in a hard-to-solder SOT-363 package.

The Texas Instruments TS5A3159-DBVR in the SOT-23 package is even lower at 1Ω and claims to have a THD of 0.005% at 4.9V peak-to-peak into a load of 600Ω.

I reckon these chips could work well in low-voltage, low-impedance portable audio circuitry. Samples are coming; it will be worth checking how they perform in the virtual earth circuit in Fig.85.

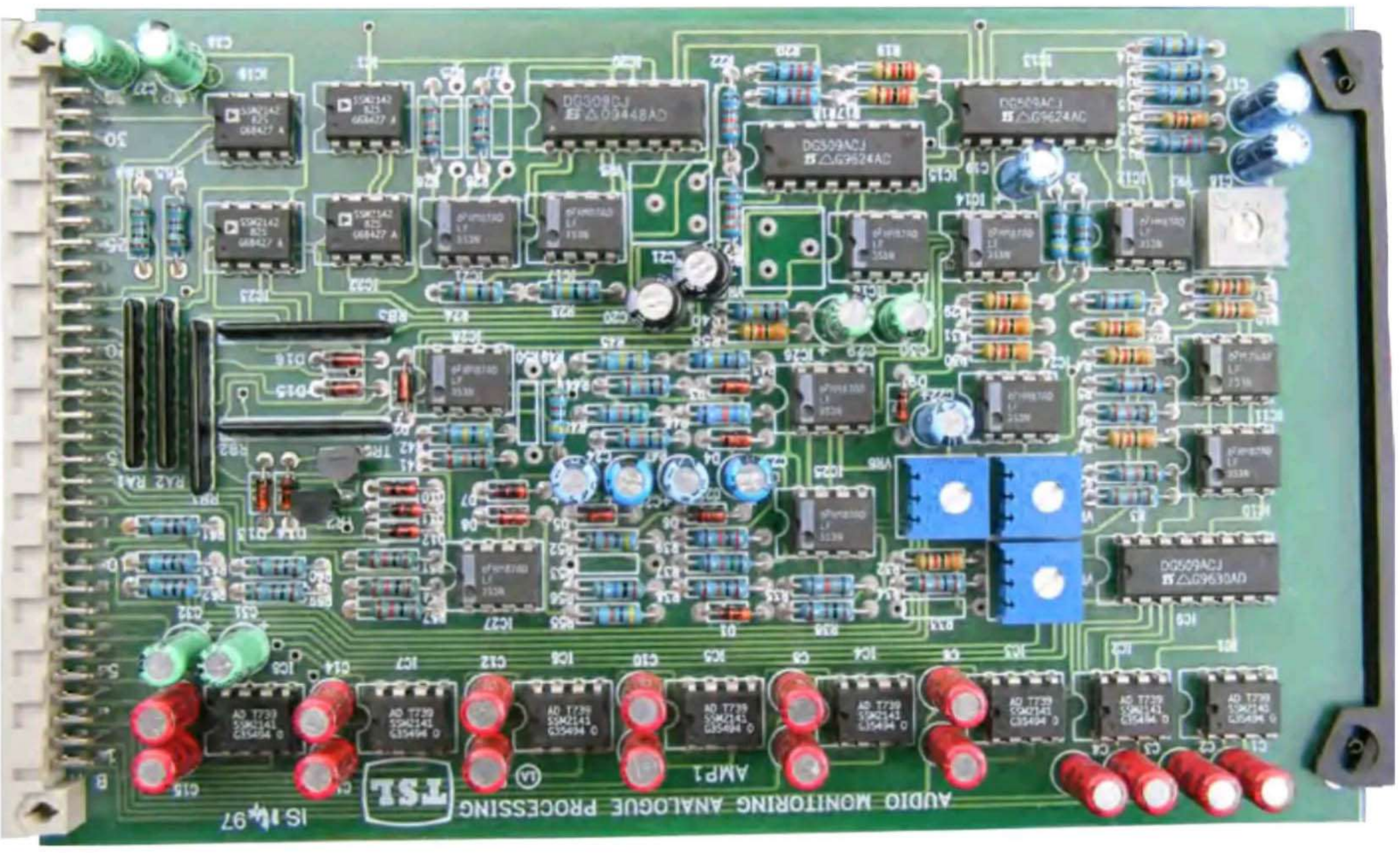


Fig.101: the TSL analog audio processor card. The DG509 switches are still made; sadly, the Analog Devices SSM2141 balanced line receivers and SSM2142 balanced line drivers are obsolete. TSL certainly splashed out on parts!

TekTrain

FREE Training Video
https://bit.ly/tektrain

Transform Your Passion into Expertise with Our Embedded Software Mentoring Programme!

Join Us Today – Perfect for Beginners, Enthusiasts, Future Professionals!

Are you passionate about making things and eager to dive deeper into Embedded Software Engineering? Whether you're a beginner, a hobbyist or looking to take your career to the next level, our programme is for you.

- Start your journey with easy-to-follow lessons and hands-on projects
- Enhance your skills with advanced topics and real-world applications
- Gain industry insights and prepare for a successful career

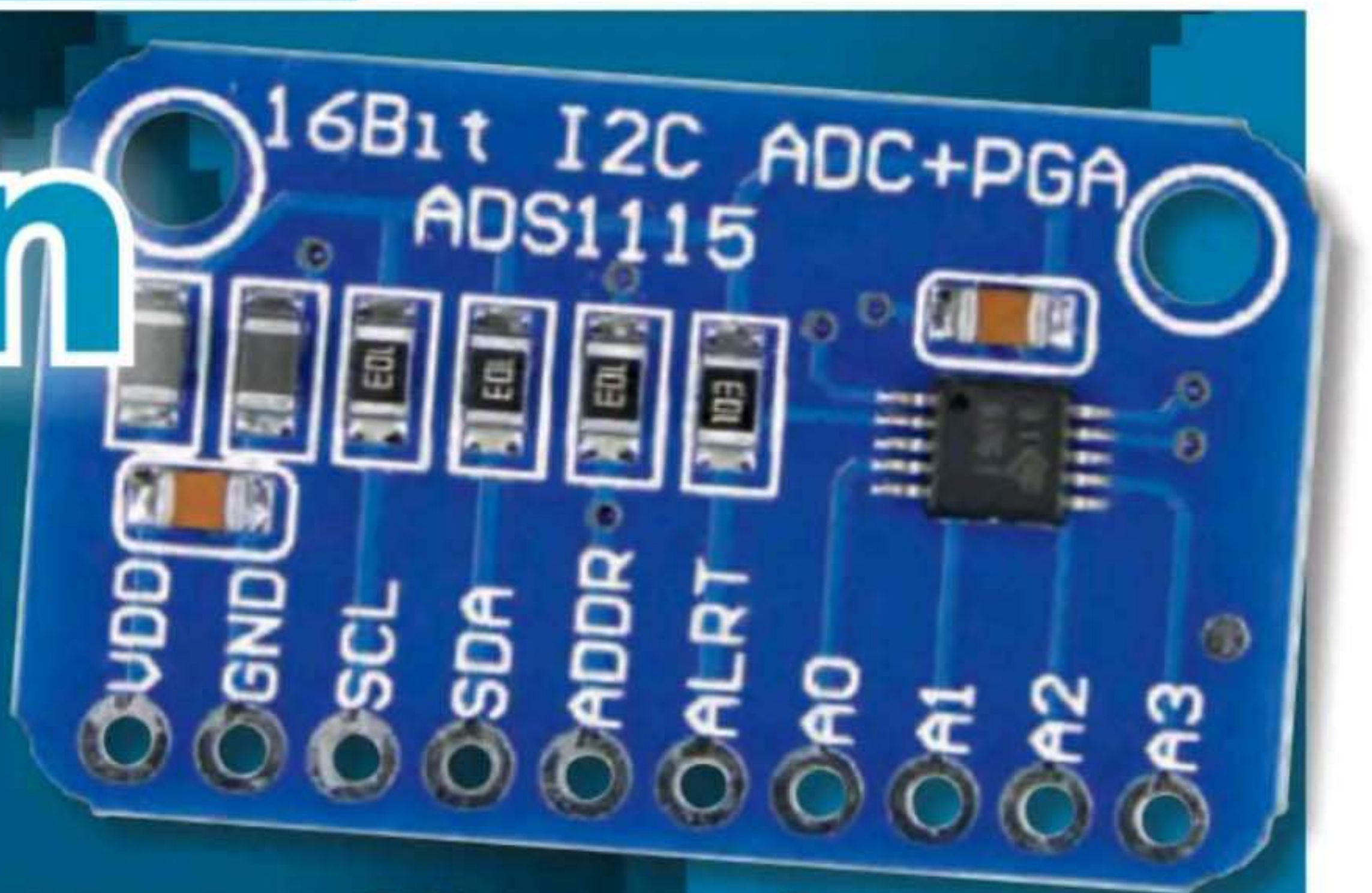
What You Get:

- An online mentorship programme from an experienced engineer
- Access to hardware, exclusive resources and tutorials
- Real-world programming experience
- Networking opportunities with others like you who are learning

Ready to Elevate Your Skills? Enroll Now!
07897-024004 | tektrain.co.uk

Using Electronic Modules with Jim Rowe

16-bit precision 4-input ADC



This month we're looking at the tiny ADS1115 that can add up to four high-speed 16-bit analog-to-digital conversion (ADC) channels to almost any microcontroller. It has a built-in I²C serial interface, so it can be easily connected to popular microcontrollers like an Arduino Uno or Nano.

Let's say you want to make precise measurements of analog voltages or currents with one of the common microcontroller units (MCUs). The ADC in most MCUs provides a resolution of only ten bits over a range of either 3.3V or 5V, which corresponds to a precision of ±1.6mV (3.3V ÷ 1024 ÷ 2) or ±2.5mV (5V ÷ 1024 ÷ 2), where 1024 is equal to 2^{10}.

That is acceptable for many applications, but not good enough if you want to make precision measurements, especially of small voltages.

That's where this module is worth considering because it allows you to add precision 16-bit ADC capability to any of the popular MCUs. As a result, you will be able to make much more precise measurements, even on quite small signals. It offers a precision improvement of 64 times compared to a 10-bit ADC or 16 times compared to a 12-bit ADC.

It is not just useful for a single-ended full-scale range of 3.3V or 5V either, because the module gives you a choice of six different full-scale ranges: ±6.144V, ±4.096V, ±2.048V, ±1.024V, ±512mV or ±256mV. This means that the smallest step size on the highest range is 187.5μV, while on the lowest range, it's 7.8125μV.

Note though that the inputs must go no more than 0.3V beyond the supply rails and those measurement ranges can be between two inputs or from an input to ground. So you can't actually measure voltages very far below ground or above the (typically 3.3V or 5V) supply range.

A further feature of the module is that it has four analog inputs, which can be used to measure either four different voltages with respect to ground, or to provide two differential inputs.

Another feature of the module which adds to its appeal is the ability to make measurements at eight different rates, from eight per second to 860 per second. It connects to the MCU via a standard two-wire I²C serial interface, with the ability to set the module's I²C port to one of four addresses: 48h, 49h, 4Ah or 4Bh (h = hexadecimal). That means you can connect up to four modules to a single I²C port on an MCU, each set for a different I²C address.

Even if you don't need the improved precision, if you've run out of ADC channels on your micro, it might be worth considering this module for the extra analog inputs it provides. If you're already using an I²C serial bus in your project, it won't even take up any more pins on the micro to add as many as 16 more analog inputs. Otherwise, you can dedicate two digital pins – not a bad swap.

In short, it's a very flexible and impressive precision ADC module. All of these capabilities are due to the IC that forms the 'heart' of the module: an ADS1115 made by Texas Instruments. So let's look at the innards of this impressive chip.

Inside the ADS1115

Fig.1 shows the basic block diagram of the ADS1115. The 16-bit delta-sigma ADC is in the centre, with the chip's built-in voltage reference just above it and the internal clock oscillator just below.

To the left of the ADC (on its input side) is a programmable-gain differential amplifier (PGA), providing the chip's six full-scale ranges.

Left of the PGA is the input multiplexer (MUX), which selects which of

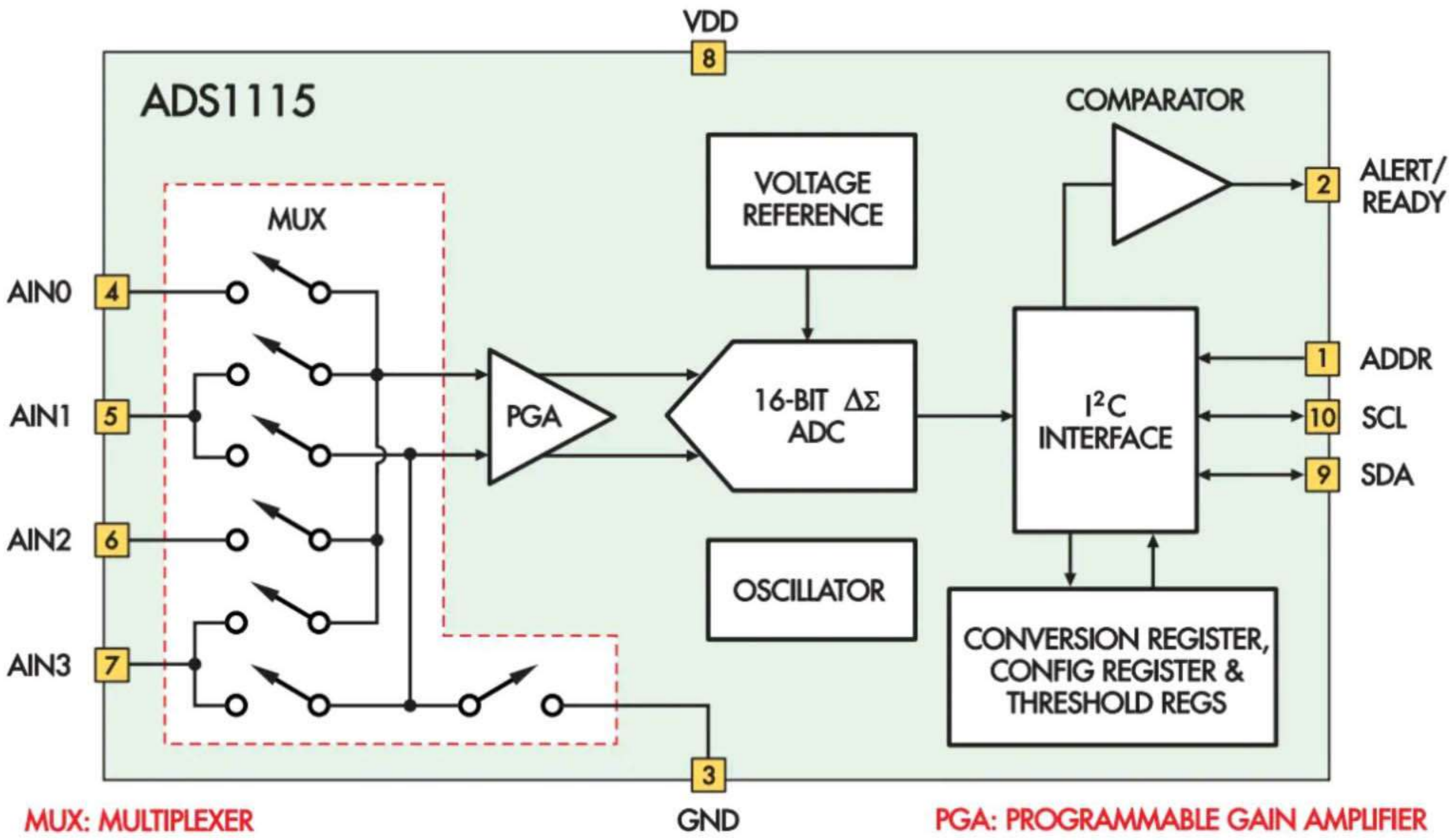


Fig.1: a block diagram of the ADS1115 IC. This shows the ADS1115 can be configured as four single-ended channels or as two differential channels.

the four single-ended inputs (AIN0-AIN3) are connected to the input of the PGA, or which two inputs are connected as a differential input.

When one of the single-ended inputs is selected, the lower input of the PGA is connected internally to ground. In contrast, when two inputs are selected for differential measurements, each is connected to one of the PGA inputs.

You can see the chip's I²C interface to the right of the ADC. This provides two-way communication between the ADS1115 and the external MCU, with programming and control data inwards, and the measurement data stream outwards.

This is done using the SCL and SDA pins; the ADDR pin sets the chip's I²C address by linking it to one of the V_{DD}, GND, SCL or SDA pins, as will be explained shortly.

Above the I²C interface is a comparator with its output connected to the chip's ALERT/READY (or ALRT/RDY) pin. The comparator can be programmed to perform one of two functions: either an alert 'flag' whenever the ADC output reaches the top or bottom threshold of its measurement range, or as an indication that a measurement has been made and the result is ready for 'collection' by the MCU.

The final section of the ADS1115 is below the I²C interface. This comprises four 16-bit registers:

1. The Conversion register, which holds the last conversion data.
2. The Configuration register, which holds the programming bits for the chip's input multiplexer, the gain/range settings for the PGA, the sampling rate setting and whether the device is to operate in single-shot or continuous conversion mode – see Fig.2.
3. Lo-thresh, the lower threshold value for the Alert Comparator.
4. Hi-thresh, the upper threshold value for the Alert Comparator.

Data in the Conversion, Lo-thresh and Hi-thresh registers is stored in signed two's complement format: from 8000h (-32768) to 7FFFh (+32767). Because this ADC has differential inputs, it can produce negative results, meaning that a signed number is needed. This also means that there are effectively 15 bits of resolution when single-ended samples are taken.

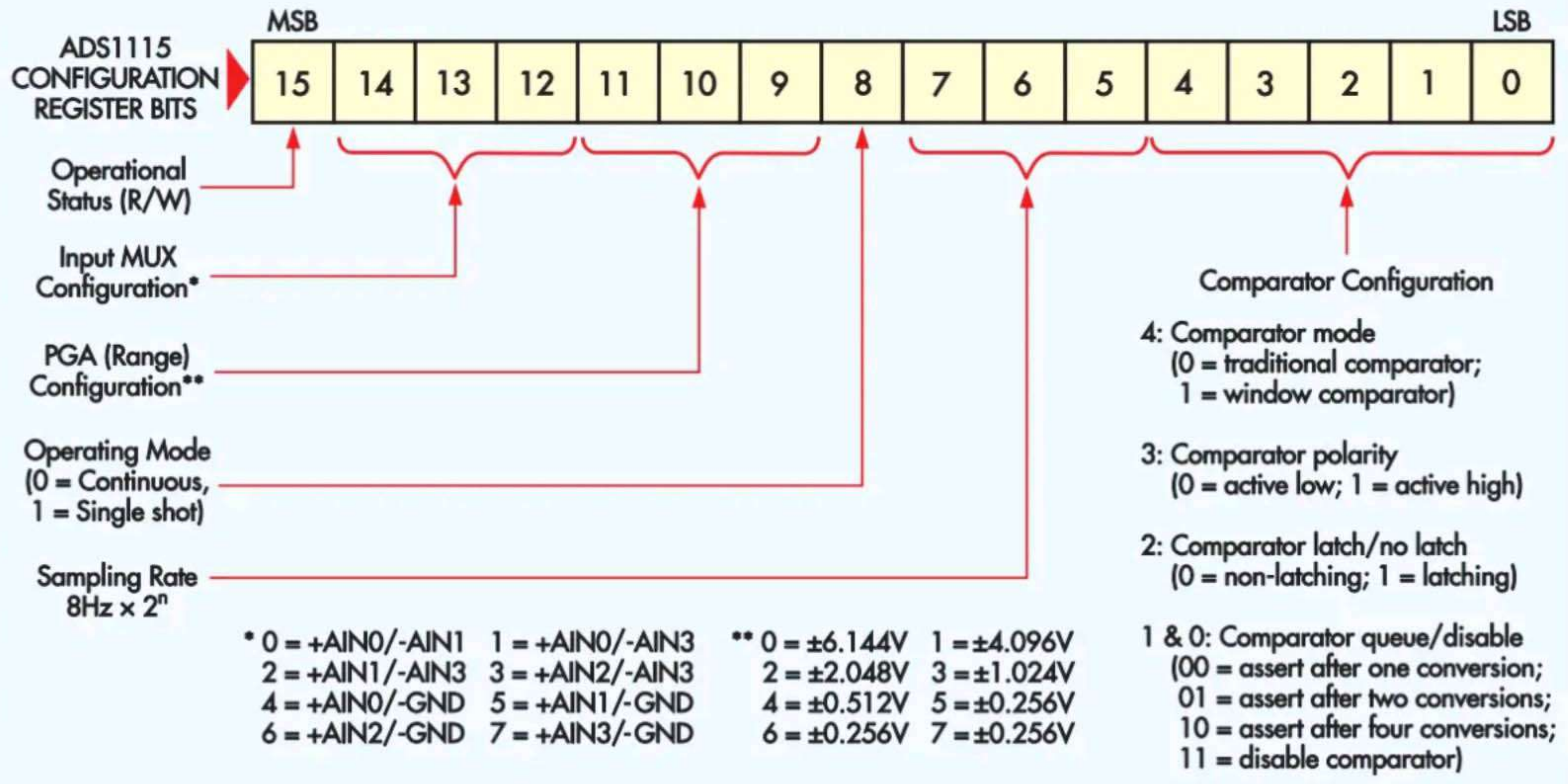

ADS1115 CONFIGURATION REGISTER BITS
MSB
LSB
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1
0
Operational Status (R/W)
Input MUX Configuration*
PGA (Range) Configuration**
Operating Mode (0 = Continuous, 1 = Single shot)
Sampling Rate 8Hz × 2ⁿ
* 0 = +AIN0/-AIN1 1 = +AIN0/-AIN3
2 = +AIN1/-AIN3 3 = +AIN2/-AIN3
4 = +AIN0/-GND 5 = +AIN1/-GND
6 = +AIN2/-GND 7 = +AIN3/-GND
** 0 = ±6.144V 1 = ±4.096V
2 = ±2.048V 3 = ±1.024V
4 = ±0.512V 5 = ±0.256V
6 = ±0.256V 7 = ±0.256V
Comparator Configuration
4: Comparator mode (0 = traditional comparator; 1 = window comparator)
3: Comparator polarity (0 = active low; 1 = active high)
2: Comparator latch/no latch (0 = non-latching; 1 = latching)
1 & 0: Comparator queue/disable (00 = assert after one conversion; 01 = assert after two conversions; 10 = assert after four conversions; 11 = disable comparator)

Fig.2: how the configuration register is arranged. This 16-bit register handles the chip's input multiplex (bits 12-14), the PGA (bits 9-11), whether the device is operated in single-shot or continuous conversion mode (bit 8) and the sampling rate (bits 5-7).

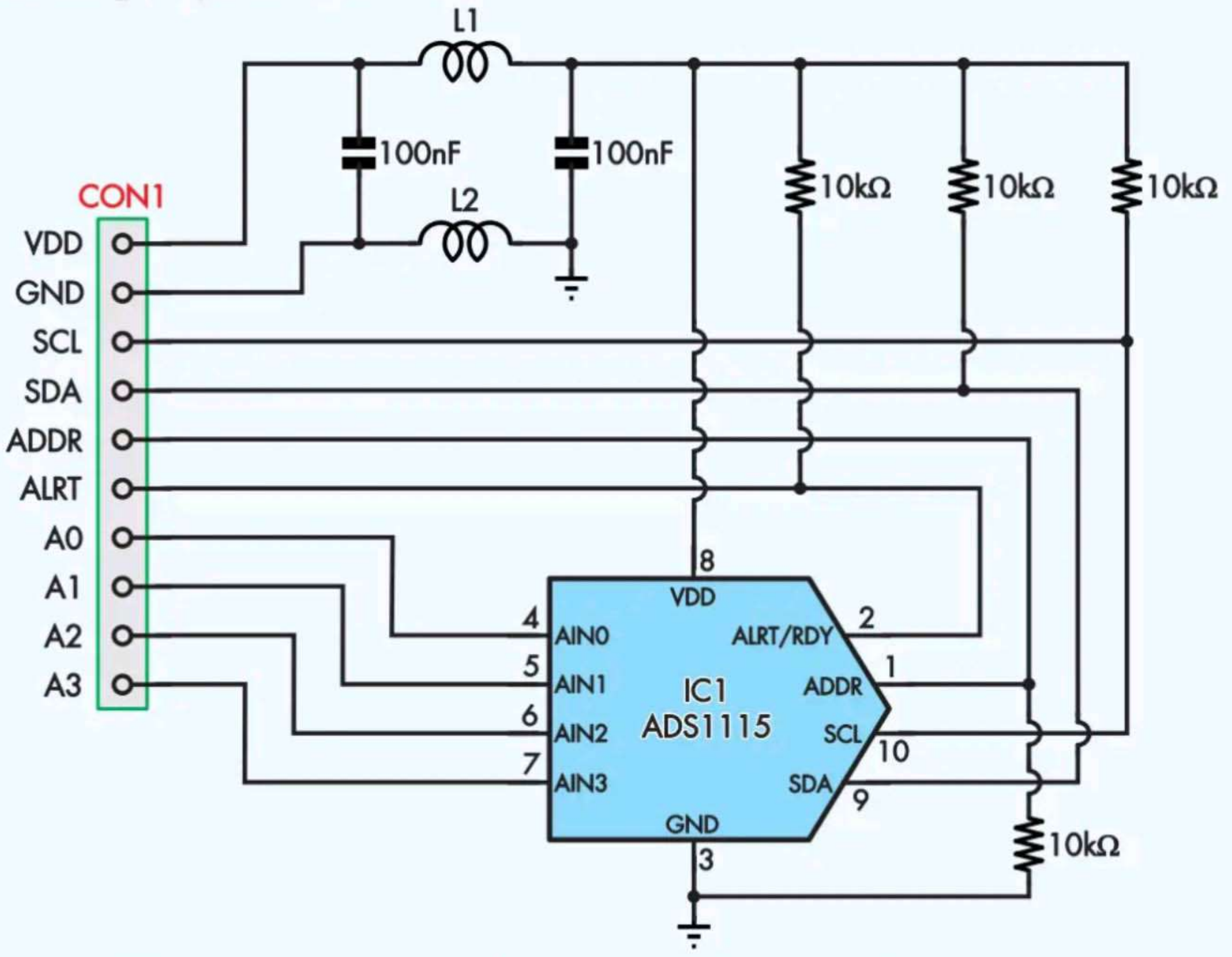

L1
L2
100nF
100nF
10kΩ
10kΩ
10kΩ
10kΩ
CON1
VDD
GND
SCL
SDA
ADDR
ALRT
A0
A1
A2
A3
IC1
ADS1115
VDD
AIN0
AIN1
AIN2
AIN3
ALRT/RDY
ADDR
SCL
SDA
GND
8
4
5
6
7
2
1
10
9
3

Fig.3: the ADS1115-based module is extremely simple, as can also be seen in the lead photo. The values of the L1 & L2 inductors are unknown, they could be ferrite beads.

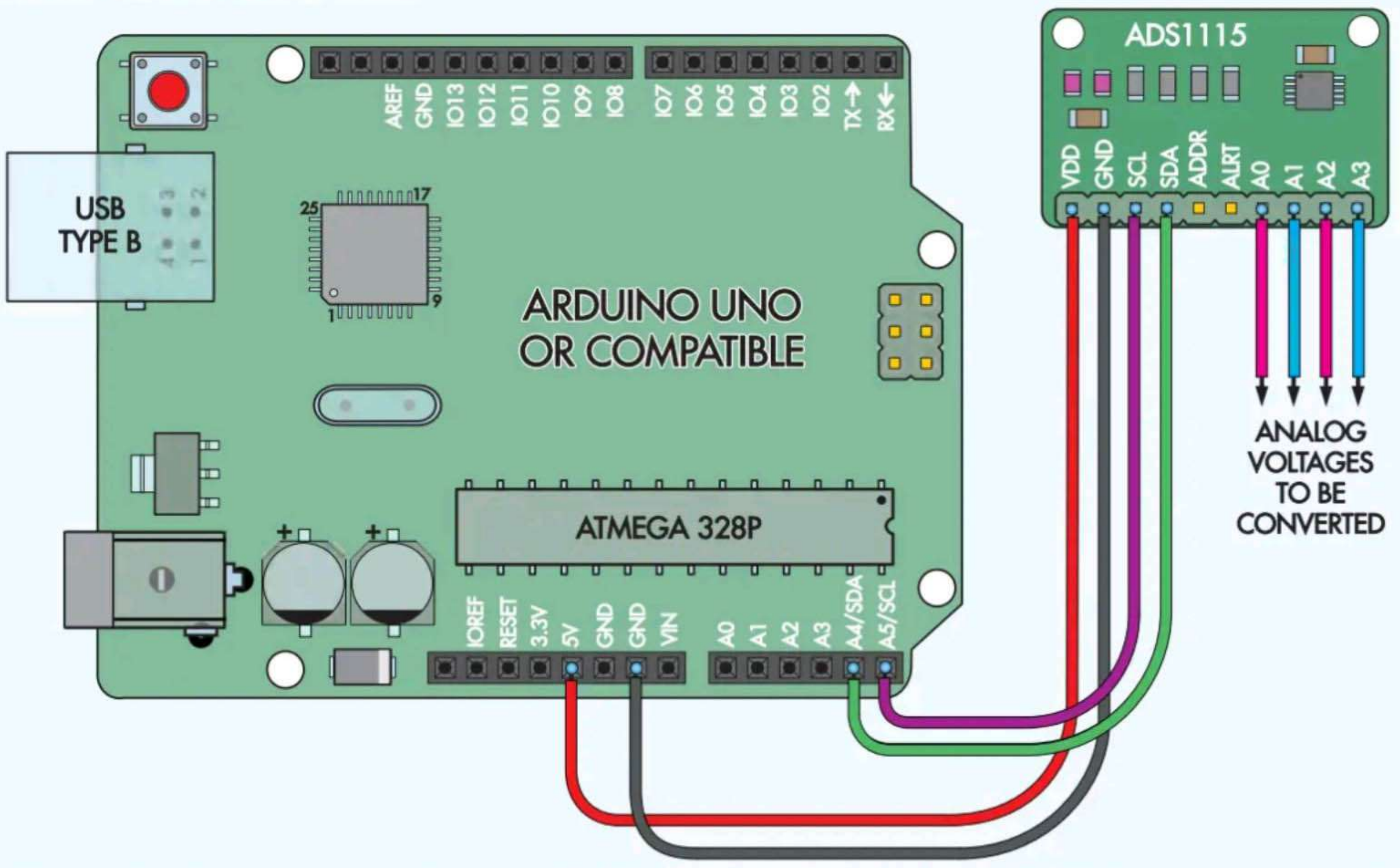

ADS1115
VDD
GND
SCL
SDA
ADDR
ALRT
A0
A1
A2
A3
USB TYPE B
ARDUINO UNO OR COMPATIBLE
ATMEGA 328P
AREF
GND
IO13
IO12
IO11
IO10
IO9
IO8
IO7
IO6
IO5
IO4
IO3
IO2
TX→
RX←
IOREF
RESET
3.3V
5V
GND
GND
VIN
A0
A1
A2
A3
A4/SDA
A5/SCL
ANALOG VOLTAGES TO BE CONVERTED

Fig.4: it's easy to connect the ADS1115-based module to an Arduino Uno or similar, as all you need to do is connect its I²C interface (SCL & SDA) and power rails to the Arduino.

As you can see, there's quite a lot inside the ADS1115's tiny (3.0 × 3.0mm) 10-pin VSSOP (very-thin shrink small-outline) package.

It can be powered from supply voltages (V_{DD}) between 2.0V and 5.5V. However, the analog input voltages must be kept within the range of (GND – 0.3V) to (V_{DD} + 0.3V) to prevent the ESD diodes inside the input MUX from conducting, which would degrade the accuracy of the ADS1115 as well as possibly damaging it.

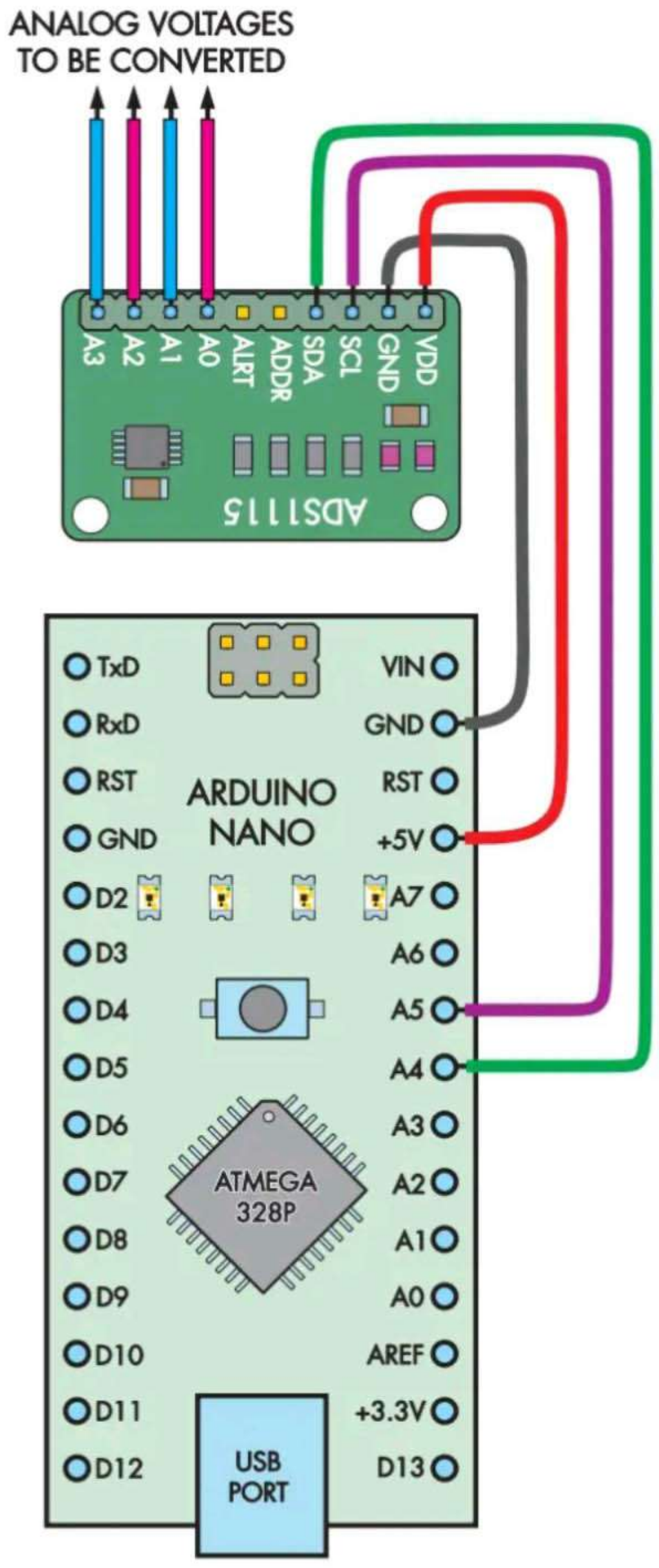


Fig.5: this wiring diagram shows how to connect the ADS1115 module to an Arduino Nano.

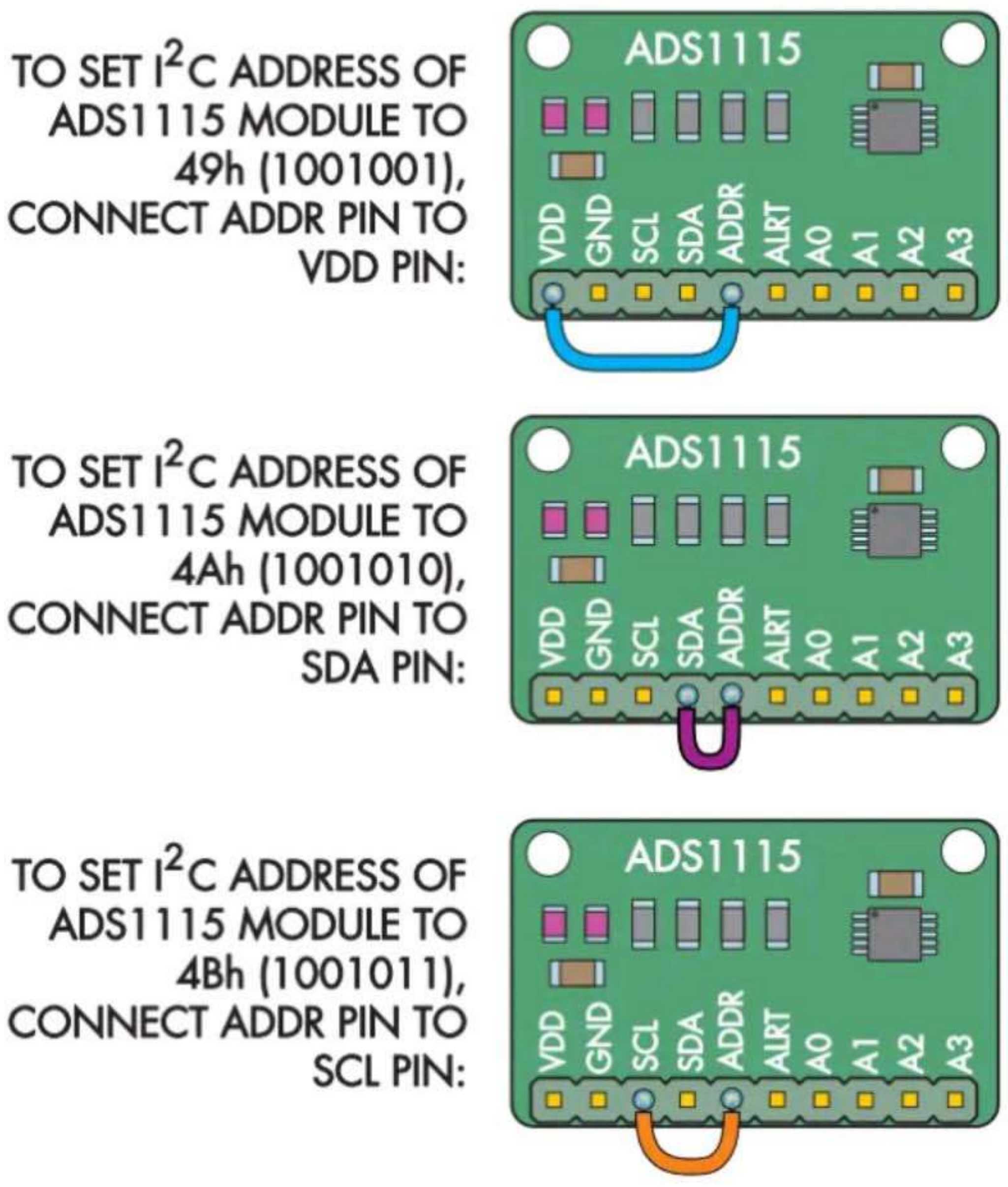


Fig.6: by default the ADS1115 has an I^{2}C address of 48h. If you plan to connect multiple ADS1115 modules to communicate with a microcontroller, then you will need to link the ADDR pin to one of the other pins as shown.

When the ADS1115 is initially powered up, it is set to its reset/default mode: the input MUX selects AIN0 and AIN1 in differential mode, the PGA is set for a full-scale range of ±2.048V, the sampling rate set to 128 samples per second and the conversion mode set to one-shot mode. If any of those need to be changed, it can be done by sending the appropriate instructions from the MCU.

The I^{2}C address of the chip is not determined by anything in the Configuration register, but by the connection to the ADDR pin.

The module circuit

As you can see from the circuit in Fig.3, there's very little in the module apart from the ADS1115 chip itself. There are three pull-up resistors connected between its SDA, SCL and ALRT/RDY pins and the V_{DD} line, a pull-down resistor between the ADDR line and GND, two small inductors (L1 and L2) of unknown value (they might even be ferrite beads), plus two 100nF capacitors which provide filtering for the module's input power.

The only other item is 10-pin SIL header CON1, which makes all the connections to the module.

Connecting to an Arduino

As mentioned earlier, one of this module's features is how its I^{2}C interface makes it easy to connect to one of the popular MCUs.

This is illustrated in Fig.4, which shows how easily it can be connected to an Arduino Uno. The module's V_{DD} pin connects to the Arduino's +5V pin, its GND pin to one of the Arduino's GND pins, its SDA pin to the Arduino's A4/SDA pin and its SCL pin to the Arduino's A5/SCL pin.

With R3 and later versions of the Uno, the last two pins can be connected to the SDA and SCL pins at upper left on the Arduino, just to the left of the AREF pin. These locations have the advantage of always being in the same position on Uno-compatible boards, regardless of which pins the micro actually uses for I^{2}C.

Connecting the module to an Arduino Nano is just as easy, as shown in Fig.5. As you can see, the module's V_{DD} pin connects to the Nano's +5V pin, the GND pin to one of the Nano's GND pins, its SCL pin to the Nano's A5 pin and its SDA pin to the Nano's A4 pin.

By default, the ADDR pin is connected to ground by a 10kΩ resistor, so the module will have the address 48h. To change it, link the module's ADDR pin to one of the pins to its left, as shown in Fig.6.

Linking this pin to the V_{DD} pin sets the I^{2}C address to 49h; linking it to the SDA sets the address to 4Ah, while linking it to the SCL pin sets the address to 4Bh.

It should be just as easy to connect the ADS1115 module to just about any other MCU, including one of the 'Mite' series (Maximite, Micromite, PicoMite etc).

Whichever MCU you want to connect the module to, you will need software to configure it and interpret its output data stream. Let's now consider how this can be done with an Arduino. This will involve finding a software library designed to communicate with the ADS1115, plus (hopefully) an example sketch to show how it's done.

Arduino software libraries

Searching the web for Arduino libraries for the ADS1115, I came up with three choices:

1. A library called ADS1x1x, written by someone named hideakitai, with the documentation and the zipped-up library code at https://github.com/hideakitai/ADS1x1x
2. A library called ADS1x15, written by Rob Tillaart, with the documentation and the library code at https://github.com/RobTillaart/ADS1x15
3. A library called ADS1115_WE, written by Wolfgang Ewald, with the documentation and the zipped-up library code at https://github.com/Wollewald/ADS1115_WE

All three come with at least one example sketch, while the third comes with no fewer than 10 examples illustrating different ways of using the ADS1115.

After trying each of these libraries and their example sketches, I decided that the ADS1115_WE library and its examples were probably the easiest to use. I also found that Mr Ewald had a very informative piece on his blog site (linked at the end of this article) giving a lot of information regarding the ADS1115, how it works and how it is programmed.

Four-channel 16-bit precision ADC

I made a few minor changes to Mr Ewald's single-shot example and tried using it to take measurements of different DC voltages.

You can see the results in the Serial Monitor listing shown in Fig.7. I had connected the test voltage to AIN1, with the AIN0 and AIN2 inputs connected to ground and the AIN3 input left floating.

The ADS1115 was programmed to set its measurement range to ±2048mV and to compare AIN1 to GND.

When I set the sketch at 7:09:36am, the voltage fed to the AIN1 input was +100mV and remained so until 7:09:49am. Then I changed it to +200mV before changing it to +500mV at 7:05:58am. Then at 7:10:07am, I changed it to +50mV, passing briefly through +100mV. Finally, at 7:10:16am, I changed the voltage to +20mV.

The AIN0 column remains fixed at readings of -0.00, as does the AIN2 column, reflecting the fact that both of these inputs were grounded. But because the AIN3 input was left floating, the readings in this column remained fixed at 0.27V.

If you decide to try out my example sketch, remember to change the address in the code to match what you have configured your module for, as shown in Fig.6.

Conclusion

Overall, this module is easy to use, flexible, and far more accurate than most microcontrollers for measuring analog voltages. It doesn't have as much precision as a good DMM, but it is nonetheless extremely handy.

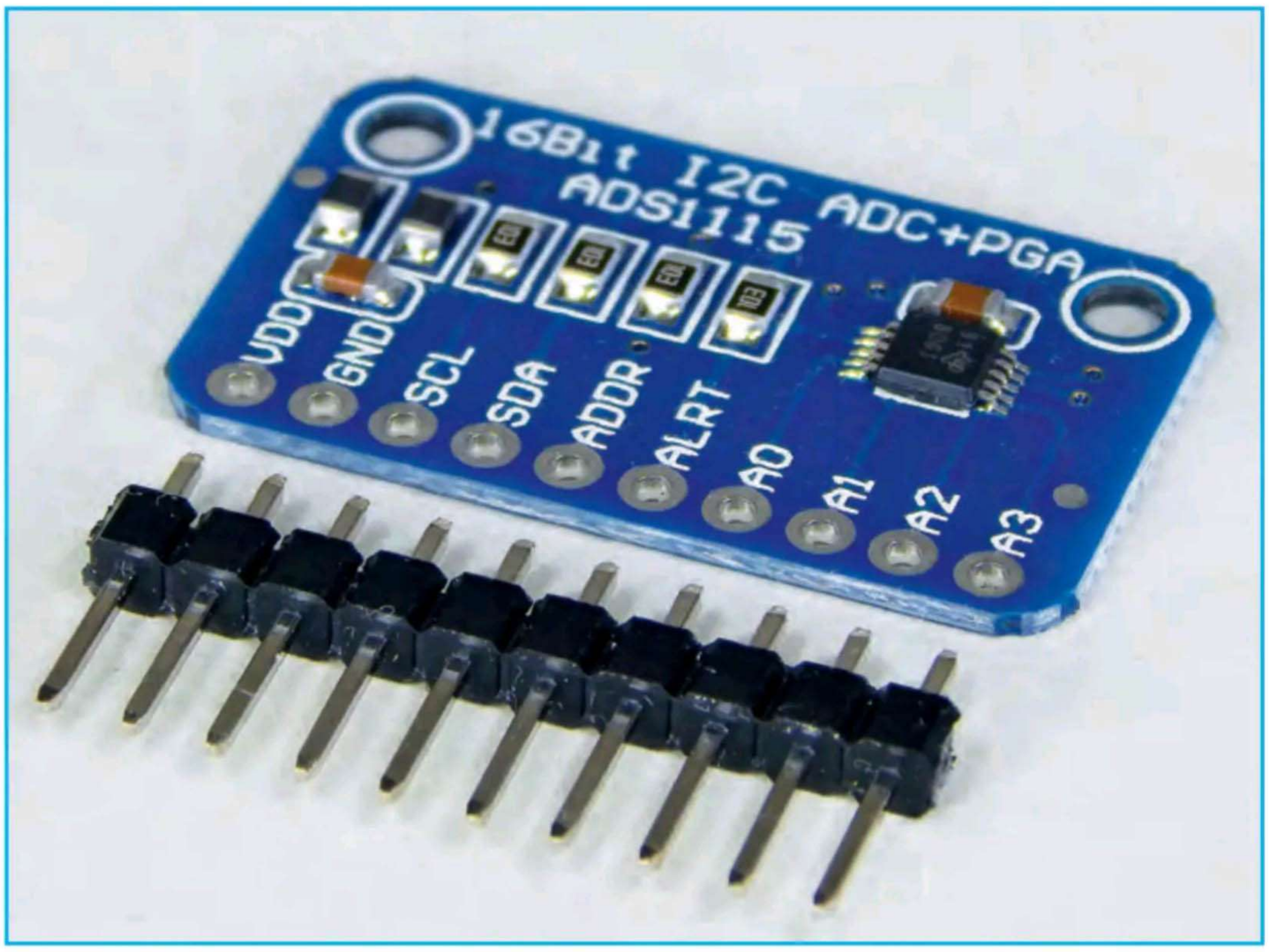
If you want to learn more about the ADS1115, you can view the datasheet at: **www.ti.com/product/ADS1115**

Also see the technical write-up on Wolles Elektronikkiste: **https://pemag.au/link/abph**

Where you can get it

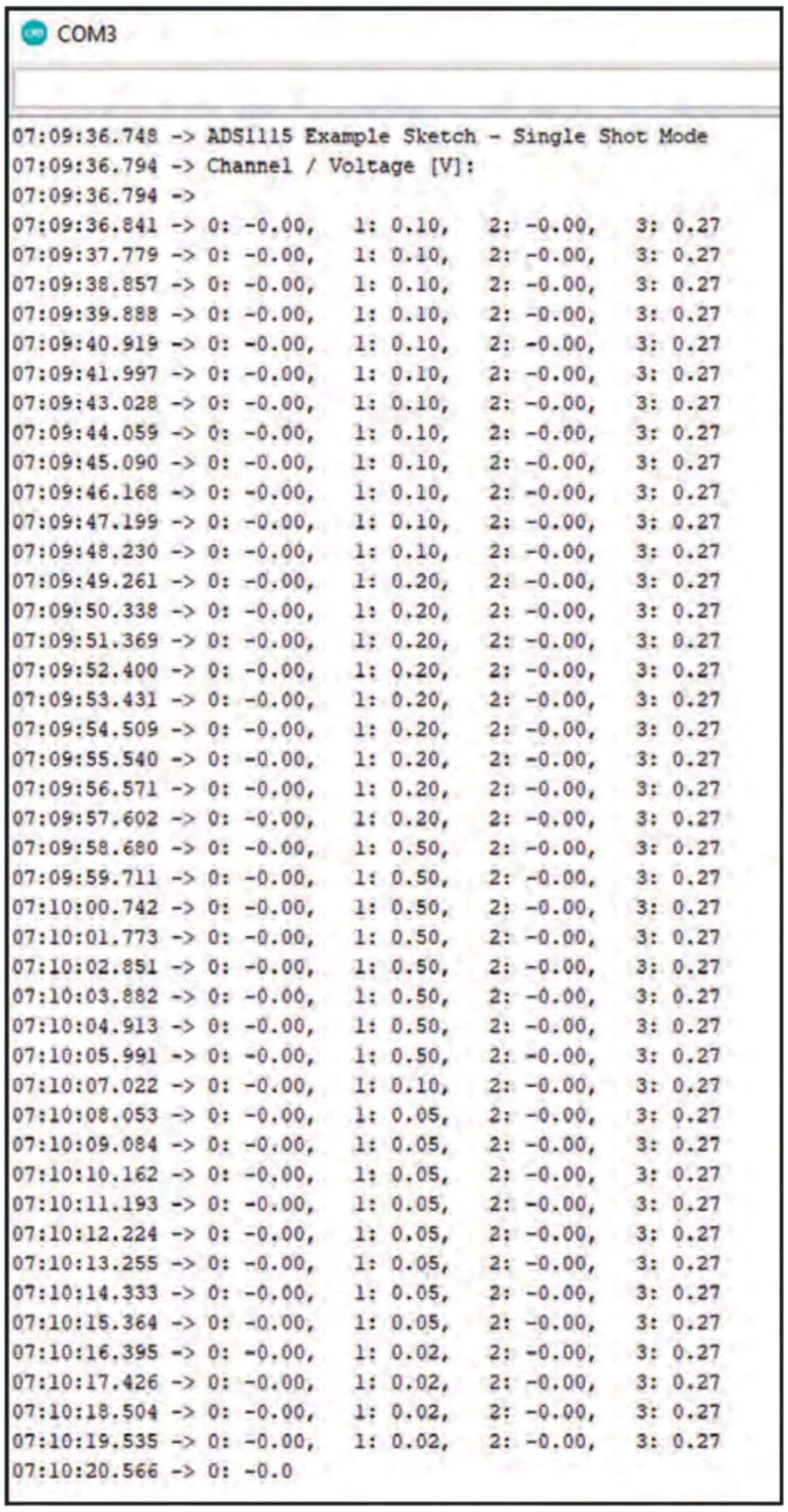
This ADS1115 module is available online from a few different suppliers. You can find multiple offers on eBay or AliExpress starting at just over £1 each (eg, **https://pemag.au/link/abyy**). Search for "ADS1115" but check the description and photos to make sure they match the module shown here.

There are some other modules using the ADS1115 that look a little different, but they are very similar in terms of circuitry. **PE**



Above: the ADS1115 module can run from a 2-5.5V supply, making it easy to use both 3.3V and 5V powered micros. The available sampling rates are: 8, 16, 32, 64, 128, 250, 475 and 860 samples per second (SPS).

Fig.7 (right): we ran Wolfgang Ewald's single-shot example code, with some minor changes, and used it to measure different DC voltages.



```
07:09:36.748 -> ADS1115 Example Sketch - Single Shot Mode
07:09:36.794 -> Channel / Voltage [V]:
07:09:36.794 ->
07:09:36.841 -> 0: -0.00,   1: 0.10,   2: -0.00,   3: 0.27
07:09:37.779 -> 0: -0.00,   1: 0.10,   2: -0.00,   3: 0.27
07:09:38.857 -> 0: -0.00,   1: 0.10,   2: -0.00,   3: 0.27
07:09:39.888 -> 0: -0.00,   1: 0.10,   2: -0.00,   3: 0.27
07:09:40.919 -> 0: -0.00,   1: 0.10,   2: -0.00,   3: 0.27
07:09:41.997 -> 0: -0.00,   1: 0.10,   2: -0.00,   3: 0.27
07:09:43.028 -> 0: -0.00,   1: 0.10,   2: -0.00,   3: 0.27
07:09:44.059 -> 0: -0.00,   1: 0.10,   2: -0.00,   3: 0.27
07:09:45.090 -> 0: -0.00,   1: 0.10,   2: -0.00,   3: 0.27
07:09:46.168 -> 0: -0.00,   1: 0.10,   2: -0.00,   3: 0.27
07:09:47.199 -> 0: -0.00,   1: 0.10,   2: -0.00,   3: 0.27
07:09:48.230 -> 0: -0.00,   1: 0.10,   2: -0.00,   3: 0.27
07:09:49.261 -> 0: -0.00,   1: 0.20,   2: -0.00,   3: 0.27
07:09:50.338 -> 0: -0.00,   1: 0.20,   2: -0.00,   3: 0.27
07:09:51.369 -> 0: -0.00,   1: 0.20,   2: -0.00,   3: 0.27
07:09:52.400 -> 0: -0.00,   1: 0.20,   2: -0.00,   3: 0.27
07:09:53.431 -> 0: -0.00,   1: 0.20,   2: -0.00,   3: 0.27
07:09:54.509 -> 0: -0.00,   1: 0.20,   2: -0.00,   3: 0.27
07:09:55.540 -> 0: -0.00,   1: 0.20,   2: -0.00,   3: 0.27
07:09:56.571 -> 0: -0.00,   1: 0.20,   2: -0.00,   3: 0.27
07:09:57.602 -> 0: -0.00,   1: 0.20,   2: -0.00,   3: 0.27
07:09:58.680 -> 0: -0.00,   1: 0.50,   2: -0.00,   3: 0.27
07:09:59.711 -> 0: -0.00,   1: 0.50,   2: -0.00,   3: 0.27
07:10:00.742 -> 0: -0.00,   1: 0.50,   2: -0.00,   3: 0.27
07:10:01.773 -> 0: -0.00,   1: 0.50,   2: -0.00,   3: 0.27
07:10:02.851 -> 0: -0.00,   1: 0.50,   2: -0.00,   3: 0.27
07:10:03.882 -> 0: -0.00,   1: 0.50,   2: -0.00,   3: 0.27
07:10:04.913 -> 0: -0.00,   1: 0.50,   2: -0.00,   3: 0.27
07:10:05.991 -> 0: -0.00,   1: 0.50,   2: -0.00,   3: 0.27
07:10:07.022 -> 0: -0.00,   1: 0.10,   2: -0.00,   3: 0.27
07:10:08.053 -> 0: -0.00,   1: 0.05,   2: -0.00,   3: 0.27
07:10:09.084 -> 0: -0.00,   1: 0.05,   2: -0.00,   3: 0.27
07:10:10.162 -> 0: -0.00,   1: 0.05,   2: -0.00,   3: 0.27
07:10:11.193 -> 0: -0.00,   1: 0.05,   2: -0.00,   3: 0.27
07:10:12.224 -> 0: -0.00,   1: 0.05,   2: -0.00,   3: 0.27
07:10:13.255 -> 0: -0.00,   1: 0.05,   2: -0.00,   3: 0.27
07:10:14.333 -> 0: -0.00,   1: 0.05,   2: -0.00,   3: 0.27
07:10:15.364 -> 0: -0.00,   1: 0.05,   2: -0.00,   3: 0.27
07:10:16.395 -> 0: -0.00,   1: 0.02,   2: -0.00,   3: 0.27
07:10:17.426 -> 0: -0.00,   1: 0.02,   2: -0.00,   3: 0.27
07:10:18.504 -> 0: -0.00,   1: 0.02,   2: -0.00,   3: 0.27
07:10:19.535 -> 0: -0.00,   1: 0.02,   2: -0.00,   3: 0.27
07:10:20.566 -> 0: -0.0
```

Max's Cool Beans

By Max the Magnificent

Arduino Bootcamp – Part 22 (more on BCD & other stuff)

As usual, my head is bursting with all sorts of "stuff" I want to talk about. For example, a young engineer recently contacted me to say he was in the process of writing a technology-related book about something or other (I don't want to give the game away). He was asking for advice regarding potential publishers.

I know how hard it can be to get one's foot through the publishing door, especially in the case of older, staid companies whose revolving doors have a habit of leaving you on the outside with your nose pressed against the glass looking in. Happily, I was able to recommend the guys and gals at No Starch Press, who bill themselves as "The finest in geek entertainment" (https://nostarch.com/).

To be honest, I cannot remember seeing a No Starch publication I didn't want to add to my library. Their *Learn to Program with Minecraft* by Craig Richardson is awesome. Some of the other No Starch tomes that grace my bookshelves are *Dead Simple Python: Idiomatic Python for the Impatient Programmer* by Jason C. McDonald, *Computer Architecture: From the Stone Age to the Quantum Age* by Charles Fox, and *The Game Console 2.0: A Photographic History from Atari to Xbox* by Evan Amos. Also, I just took delivery of *Writing a C Compiler: Build a Real Programming Language from Scratch* by Nora Sandler.

Now, before we expand on our previous discussions and take a deeper dive into the binary-coded decimal (BCD) waters, a couple of other topics are bouncing around my old noggin. So, if you will be kind enough to indulge me…

Feeling grey or gray?

The fact that some words have different spellings in British English (I hail from the UK) and American English (I currently hang my hat in the USA) can be a tad tiresome. For example, 'grey' is more common in British banter, while 'gray' is more frequent in American argot.

By some strange quirk of fate, this leads us to the topic of Frank Gray (1887–1969), who was a physicist and researcher at Bell Labs. Frank was responsible for numerous innovations, especially in the field of television.

However, he is primarily remembered for coming up with the concept of Gray codes (when I started out, I wondered why they weren't called "Grey codes").

Compare and contrast the two 3-bit sequences shown in Fig.1, for example. In the case of transitions between 1 and 2 or 5 and 6 in decimal, two bits change when using a standard binary counting sequence. Similarly, in the case of transitions between 3 and 4 or 7 and 0 in decimal, all three bits change in standard binary. By comparison, the ordering of the values in the Gray code are such that adjacent values always differ by only a single bit.

In addition to their use when arranging the rows and columns in Karnaugh maps, Gray codes are widely employed to prevent spurious outputs from electromechanical switches and to facilitate error detection and correction in digital communications.

Count on me

As that philosopher for our time, Mickey Mouse (1928–) famously said:

Decimal	Binary	Gray
0	0 0 0	0 0 0
1	0 0 1	0 0 1
2	0 1 0	0 1 1
3	0 1 1	0 1 0
4	1 0 0	1 1 0
5	1 0 1	1 1 1
6	1 1 0	1 0 1
7	1 1 1	1 0 0

Fig.1: Binary vs. Gray codes.

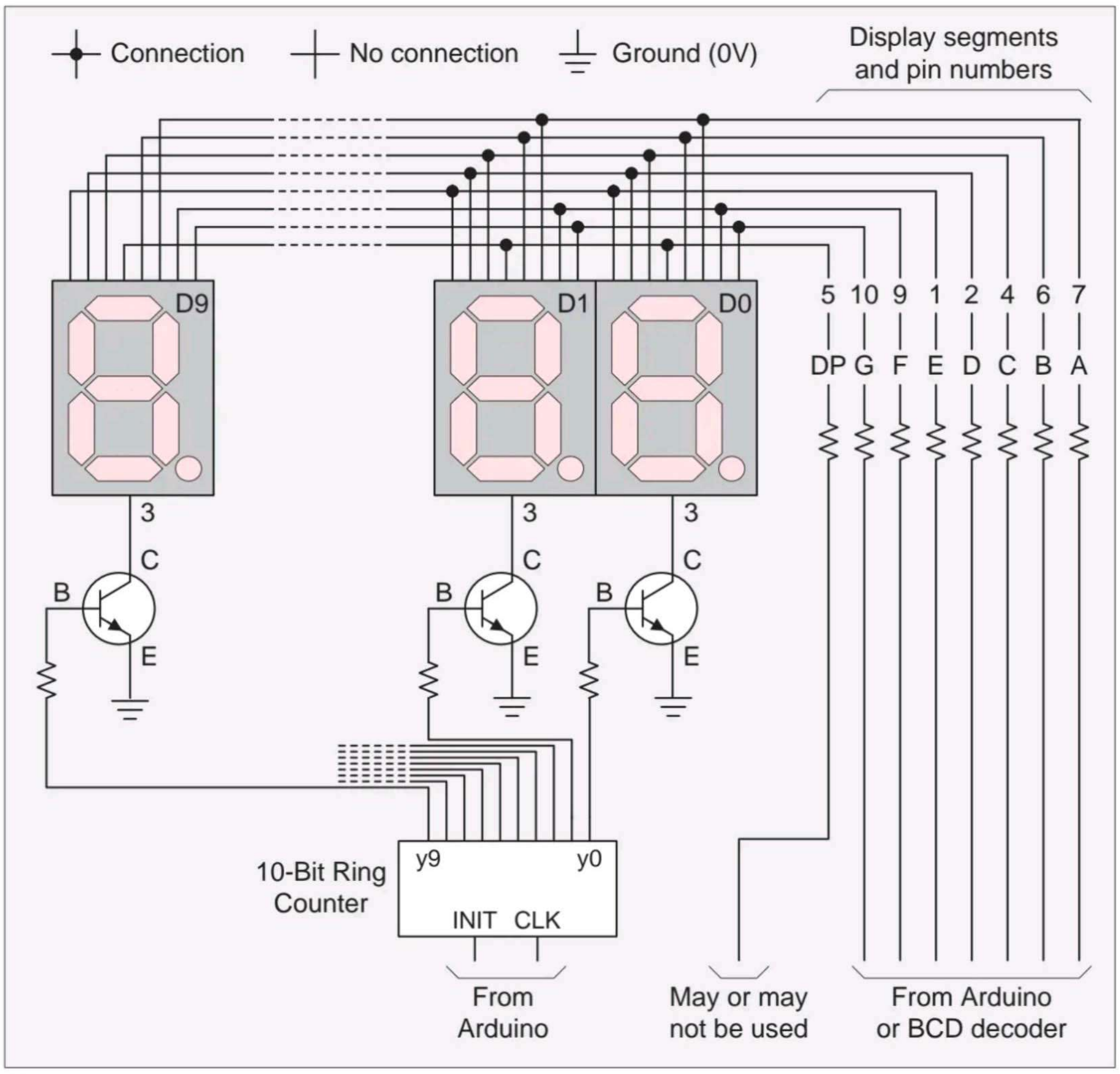


Fig.2: Using a 10-bit ring counter to control ten 7-segment displays.

"Arithmetic is being able to count up to twenty without taking off your shoes". You can't argue with logic like that.

In my last column, we discussed how we could use ten D-type flip-flips to implement a 10-bit ring counter. The idea is that only one bit from this counter is active (logic 1) at any time, so clocking the counter results in a sequence of 1000000000, 0100000000, 0010000000… 0000000100, 0000000010, 0000000001, and back to 1000000000 again.

As illustrated in Fig.2, we could use the outputs from our ring counter to control ten transistors. In turn, these transistors could control ten 7-segment displays.

Personally, I think this is rather clever. When used in conjunction with a BCD decoder, it means we can control all ten 7-segment displays using just six of the Arduino's pins. Is there any way we could increase our cleverness quotient? Well, by golly, I'm glad you asked because there certainly is.

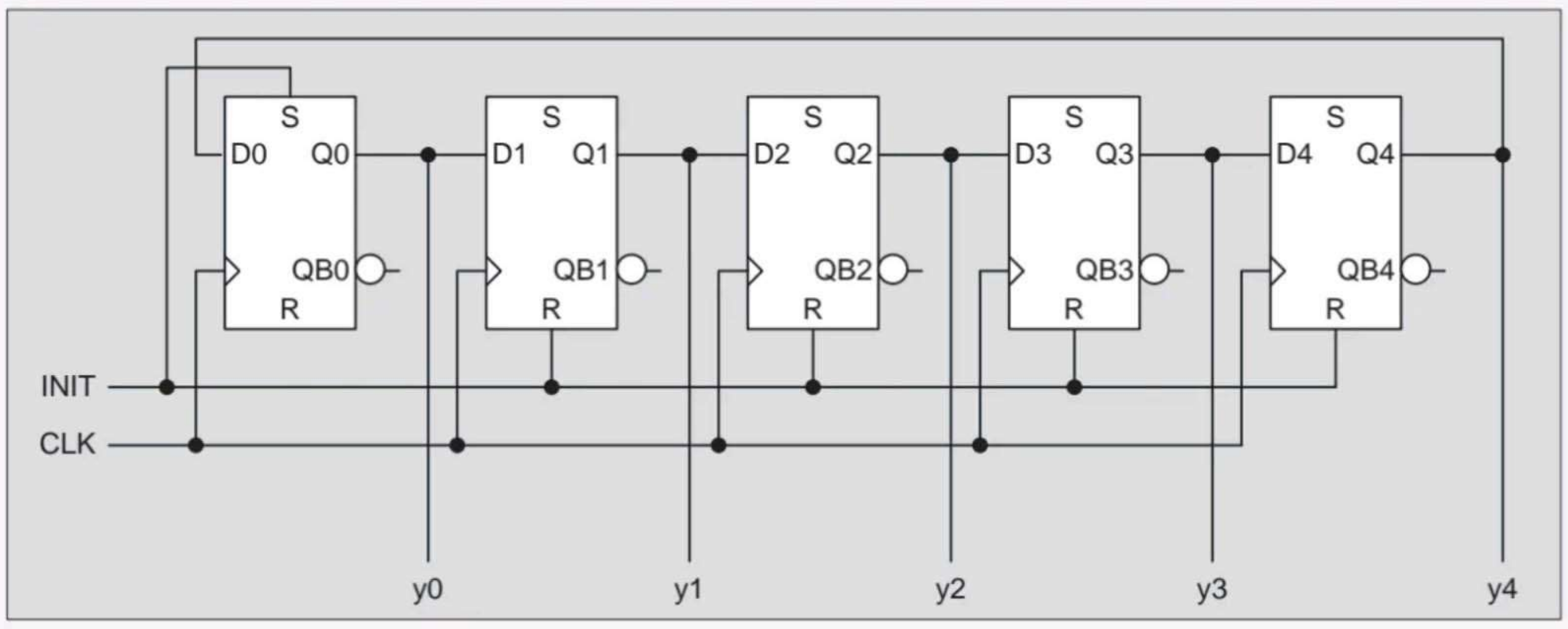
Jolly Japes with Johnsons

I love digital logic. I stand in awe of people like Frank Gray who can come up with logical concepts like Gray codes – things I wouldn't be capable of conceiving myself in a thousand years. Another of my heroes is the American inventor, engineer, computer pioneer and professor, Robert Royce "Bob" Johnson (1928–2016). Amongst many other inventions and innovations, Bob came up with something we now call the Johnson counter.

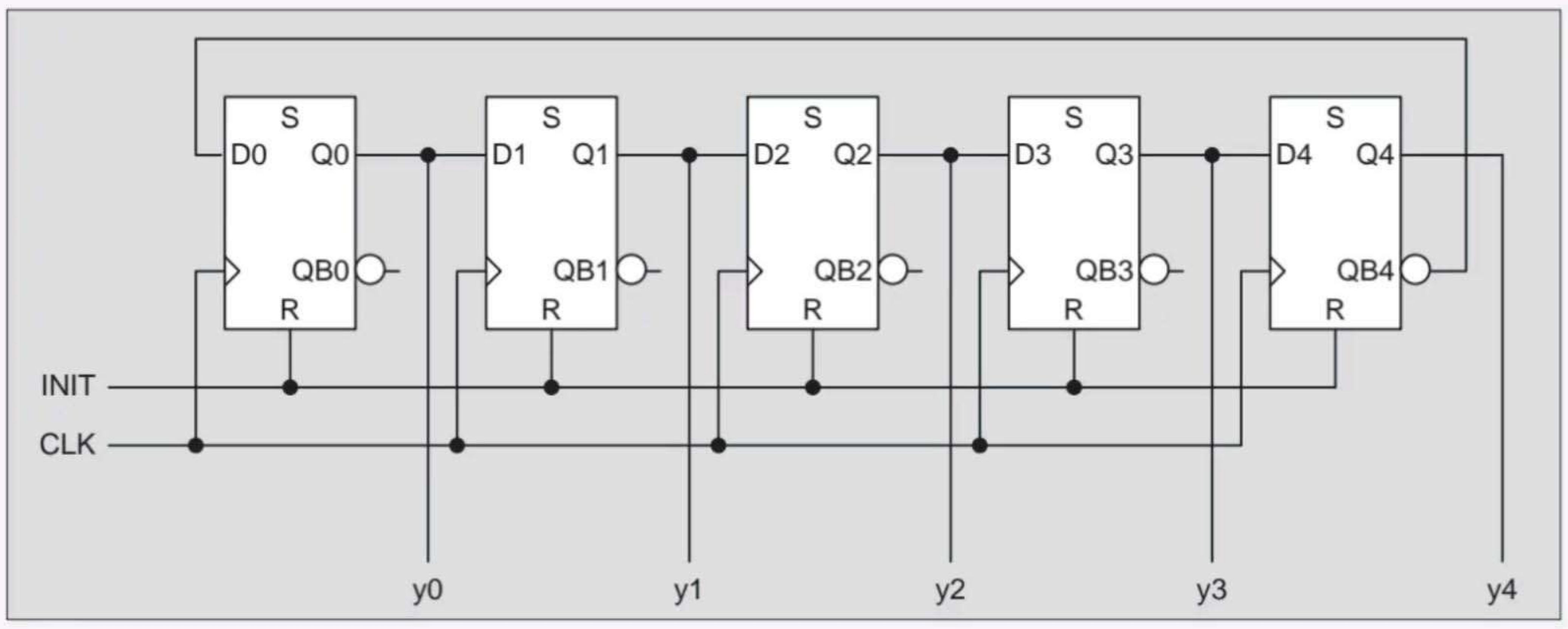
The ring counter we've been considering thus far is a 'straight ring counter'. It may also be referred to as a 'one-hot counter' because only one output is active, or 'hot', at any time. A comparison of 5-bit (aka 5-stage) straight ring and Johnson counters is shown in Fig.3. Can you spot the two big differences between these implementations?

The first difference derives from the wiring of the `INIT` signal. In the case of the straight ring counter, this signal is connected to the `S` (set) input of the first stage and the `R` (reset) inputs of the remaining stages. This means that when the `INIT` signal is placed in its active state, our 5-bit straight ring counter will be loaded with a value of 10000. By comparison, in the case of the Johnson counter, this signal is connected to the `R` inputs of **all** the stages. As a result, when the `INIT` signal is placed in its active state, our 5-bit Johnson counter will be loaded with a value of 00000.

The second difference occurs in the way in which we feed the inputs to our counters. In the case of the straight ring counter, the `D0` input to the first stage is fed from `Q4`, which is the **true** output from the final stage. By comparison, in the case of the Johnson counter, the `D0` input is fed from `QB4`, which is the **false** output from the final stage.



(a) 5-stage straight ring counter



(b) 5-stage Johnson counter

Fig.3: 5-bit straight ring vs. Johnson counter implementations.

Straight Ring Counter					
State	Q0	Q1	Q2	Q3	Q4
0	1	0	0	0	0
1	0	1	0	0	0
2	0	0	1	0	0
3	0	0	0	1	0
4	0	0	0	0	1
0	1	0	0	0	0
1	0	1	0	0	0
2	0	0	1	0	0
3	0	0	0	1	0
4	0	0	0	0	1
0	1	0	0	0	0

Johnson Counter					
State	Q0	Q1	Q2	Q3	Q4
0	0	0	0	0	0
1	1	0	0	0	0
2	1	1	0	0	0
3	1	1	1	0	0
4	1	1	1	1	0
5	1	1	1	1	1
6	0	1	1	1	1
7	0	0	1	1	1
8	0	0	0	1	1
9	0	0	0	0	1
0	0	0	0	0	0

Fig.4: 5-bit straight ring vs Johnson counter sequences.

Imagine taking a strip of paper and joining its ends to form a circular band. In this case, the band will have two faces or sides (inside and outside) and two edges. This would be the equivalent to our straight ring counter. If we take another strip of paper and give one of its ends a one-half twist before joining the two ends, we end up with a Möbius strip, which boasts only one side and one edge. This would be the counterpart to our Johnson counter.

The way in which we implement the feedback path in our Johnson counter explains why this form of counter may also be referred to as a 'twisted ring counter', a 'switch-tail ring counter', a 'walking ring counter', or a 'Möbius counter' (good luck finding tidbits of trivia like this in other, lesser electronic hobbyist magazines!).

So, how do these seemingly trivial differences affect the outputs of these counters in the real world? The easiest way to wrap our brains around this is to compare the number of count states along with their corresponding values as proffered by these dueling implementations (Fig.4).

As we see, there are multiple differences in the sequences. A glance at the highlighted 1s in Fig.4 reveals that the straight ring counter circulates a single 1 bit around the ring (it could be a 0 if we wished). By comparison, the Johnson counter circulates a stream of 1s followed by a stream of 0s.

A by-product of the previous point is that, in the case of the straight ring counter, two bits always change when transitioning from one state to the next. By comparison, in the case of the Johnson counter, only a single bit changes when transitioning from one state to the next. This means the Johnson counter

Johnson Counter						Johnson Decade Counter
State	Q0	Q1	Q2	Q3	Q4	Decoded Outputs
0	0	0	0	0	0	y0 = Q0B & Q1B & Q2B & Q3B & Q4B
1	1	0	0	0	0	y1 = Q0 & Q1B & Q2B & Q3B & Q4B
2	1	1	0	0	0	y2 = Q0 & Q1 & Q2B & Q3B & Q4B
3	1	1	1	0	0	y3 = Q0 & Q1 & Q2 & Q3B & Q4B
4	1	1	1	1	0	y4 = Q0 & Q1 & Q2 & Q3 & Q4B
5	1	1	1	1	1	y5 = Q0 & Q1 & Q2 & Q3 & Q4
6	0	1	1	1	1	y6 = Q0B & Q1 & Q2 & Q3 & Q4
7	0	0	1	1	1	y7 = Q0B & Q1B & Q2 & Q3 & Q4
8	0	0	0	1	1	y8 = Q0B & Q1B & Q2B & Q3 & Q4
9	0	0	0	0	1	y9 = Q0B & Q1B & Q2B & Q3B & Q4

Fig.5: Decoding 10 one-hot signals from a 5-bit Johnson counter.

is implementing a form of Gray code.

And, speaking of states, the 5-bit straight ring counter passes through only five states (numbered 0 to 4) before returning to its starting state. We could generalize this to say that an n-bit straight ring counter passes through n states before returning to its starting state.

By comparison, the 5-bit Johnson counter passes through ten states (0 to 9) before returning to its starting state. We could generalize this to say that an n-bit Johnson counter passes through 2n states before returning to its starting state.

We've already seen how to control ten 7-segment displays using a 10-bit straight ring counter. The cool thing is that, since we get both the `Q` (true) and `QB` (false) outputs from each register stage 'for free', we can achieve the same thing using a 5-bit Johnson counter in conjunction with ten 5-input AND gates, as illustrated in Fig.5.

When the states of 5-bit Johnson counter are decoded to generate ten one-hot outputs, the resulting implementation is known as a Johnson Decade Counter. These are available as off-the-shelf chips, such as the 74HC4017 (**https://pemag.au/link/abzh**).

Why go to all this trouble? When Johnson originally patented this concept in 1954, he was working with diodes, resistors, and vacuum tubes (valves). Each of his register states was implemented using two back-to-back tubes. So, instead of using 20 expensive tubes to build a 10-bit straight ring counter, Johnson required only 10 tubes to build his 5-bit counter. Furthermore, each of his ten 5-input AND gates was implemented cheaply and cheerfully using only one resistor and five diodes. Pretty clever, eh?

While this discussion isn't particularly relevant to what we aim to achieve, in my experience, being aware of concepts like this can be unexpectedly useful when one least expects it.

How common!

Before returning to the BCD fray, we must take one more little diversion (so unlike me, I know); some of the things we are going to talk about shortly can be hard to wrap one's brain around, it behooves us to take things step-by-step.

Throughout this epic Arduino Bootcamp series, we've been working with common-cathode (CC) 7-segment displays, as shown in Fig.6(a). This means the cathodes associated with the eight light-emitting diode (LED) segments are joined inside the device. We use the anodes to control the LEDs on an individual basis.

It's also possible to get common-anode (CA) 7-segment displays, shown in Fig.6(b). In this case, the anodes are internally joined and we use the cathodes to control the LEDs individually.

If you cast your mind back to our early experiments with our CC displays, you will recall that we connected our common cathode to ground and used our Arduino to control the anodes (via current-limiting resistors), as in Fig.7(a).

As we discussed way back in the mists of time (*PE*, March 2023), `LOW` and `HIGH` in our programs are seen by the compiler as being the same as 0 and 1, respectively. When we write a `LOW` (or 0) to a digital pin, this will be presented as 0V to the outside world. When we write a `HIGH` (or 1) to a digital pin, this will be presented as 5V to the outside world (at least, in the case of the Arduino Uno we're using).

In the case of our CC displays, presenting 0V (`LOW`) or 5V (`HIGH`) to an anode will switch that segment off or on, respectively. This explains why we've been using the following definitions in our programs:

```
// Definitions for CC display
#define SEG_ON    HIGH
#define SEG_OFF    LOW
```

Later, in the body of our programs, when we called the `digitalWrite()` function to instruct one of the Arduino's pins to switch a segment on or off, we used our `SEG_ON` and `SEG_OFF` definitions as arguments to the function.

Have you ever wondered why we bothered to create and use these definitions in the first place? Since we intend to use only CC displays in this series, you might feel this is redundant. There are two reasons we typically do it this way.

First, even if we only ever use CC displays, the `SEG_ON` and `SEG_OFF` names are easy for others to understand if they read our programs (and for us to

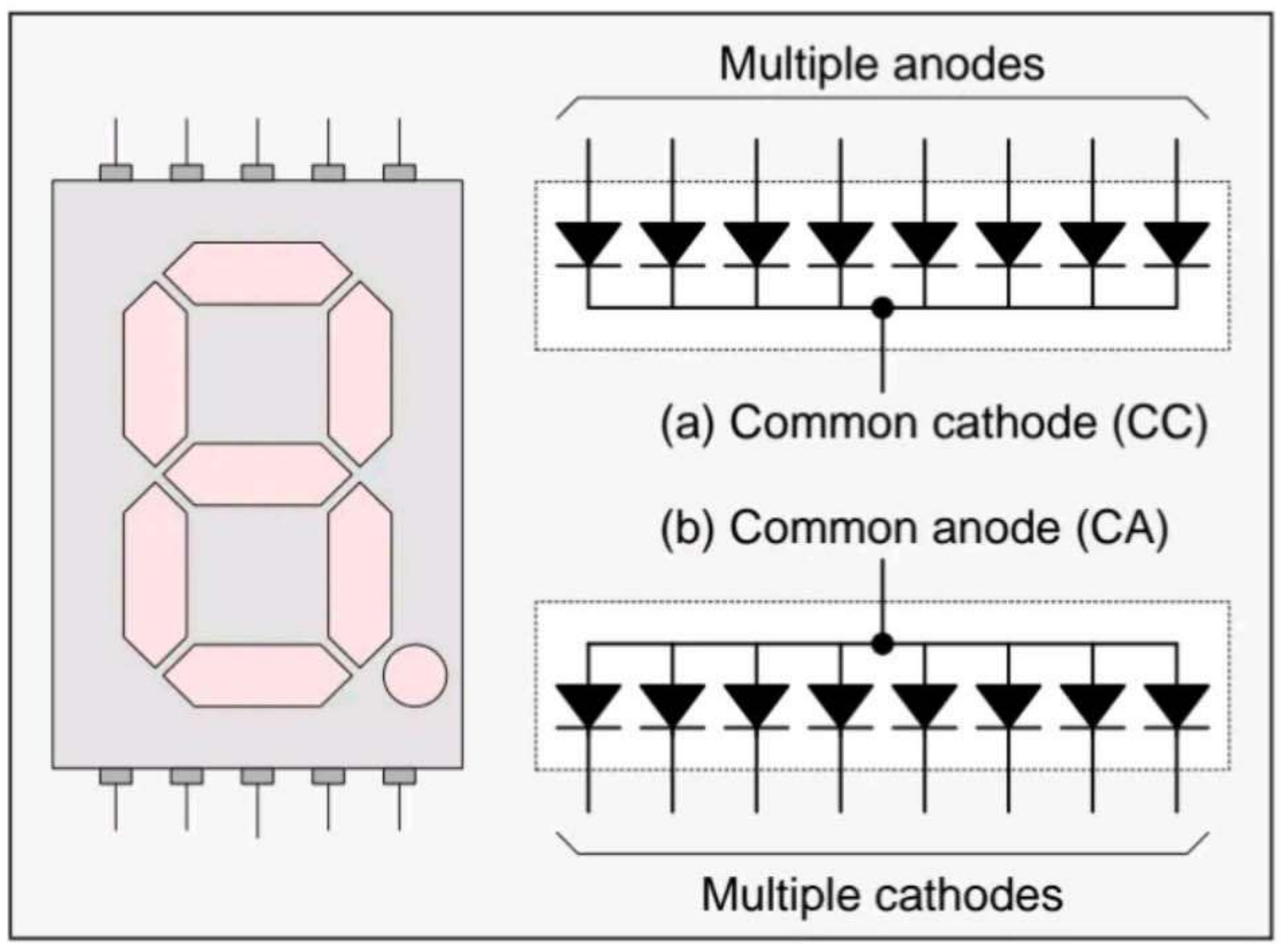
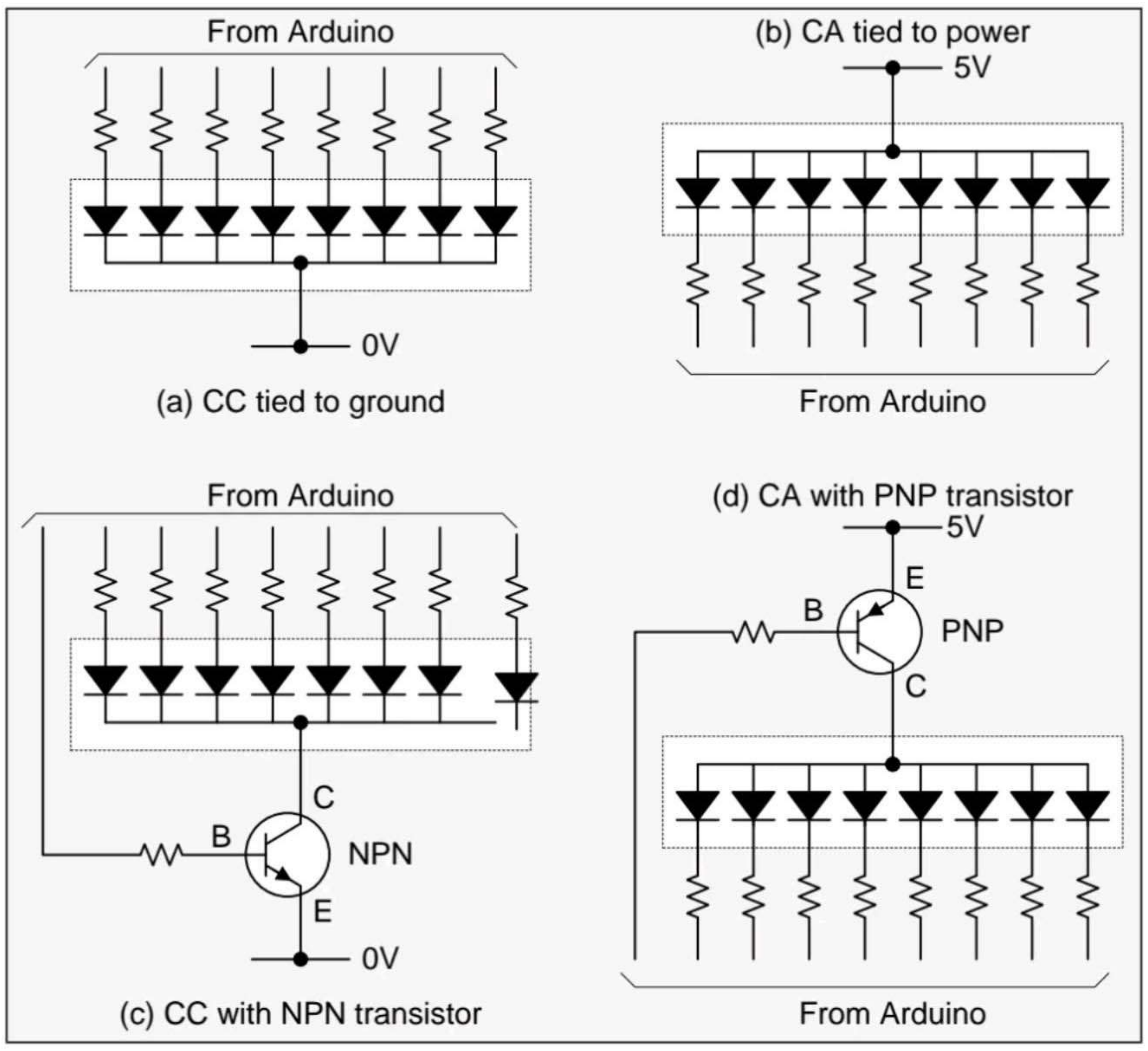


Fig.6: Common cathode vs common cathode displays.



(a) CC tied to ground

(c) CC with NPN transistor

Fig.7: CC and CA displays with and without transistors.

understand when we return long after we've forgotten how everything works).

Second, if we ever decided to exchange our displays for their pin-compatible CA counterparts, as in Fig.7(b), all we have to do on the hardware side would be to disconnect the common pin from 0V (ground) and reconnect it to 5V (power). Meanwhile, on the software side, we just need to modify our definitions as follows:

```
// Definitions for CA display
#define SEG_ON    LOW
#define SEG_OFF  HIGH
```

The rest of the program will continue to work 'as is'. Brilliant!

Switcheroo

Before we proceed, let's take a moment to remind ourselves that pure silicon is an insulator. We can modify the silicon using a process called 'doping' to add small quantities of different elements into its crystalline matrix. We can create P-type (positive) and N-type (negative) silicon, both of which **do** conduct.

The interesting stuff happens at a PN junction, where P-type and N-type materials meet. A single PN junction forms a diode, a component that only conducts in one direction. If we use the right materials, we end up with a diode that emits light when it's conducting, ie, a light-emitting diode (LED).

If we create a silicon sandwich from two back-to-back PN junctions, we have a bipolar junction transistor (BJT). These components can be used in two ways: as analog amplifiers or as digital switches.

Furthermore, there are two ways in which we can create our silicon sandwiches: NPN or PNP (all of this is explained in ~~excruciating~~ exhilarating detail in my book, *Bebop to the Boolean Boogie* (https://pemag.au/link/abzi).

Returning to our common-cathode implementation, in our more recent experiments, we've inserted an NPN transistor between the common cathode pin on our 7-segment display and the 0V (ground) rail, as in Fig.7(c). This transistor acts as a switch to simultaneously enable or disable all the segments the Arduino is trying to switch on.

The transistor we opted to use was a BC337 because this can handle up to 800mA, much more than maximum current (8 × 20mA = 160mA) if all eight segments are active at the same time.

First, we used one of these transistors to control the brightness of a single 7-segment display by switching its segments on and off very quickly while controlling the ratio of the on-to-off times (*PE*, January 2024). Next, we used two of these transistors to multiplex between two 7-segment displays (*PE*, April and May 2024). More recently (starting in *PE*, June 2024), we moved to using four transistors to multiplex between four 7-segment displays.

In our circuits, the signals driving our transistors' base (B) control terminals come (via current-limiting resistors) from the Arduino. In the case of an NPN transistor used, as in Fig.7(c), 0V on its base will switch it off (like opening a switch), thereby disconnecting the display's common cathode pin from the 0V rail and ensuring all the display segments are off.

By comparison, a 5V signal on the base resistor will switch the transistor on (like closing a switch), thereby connecting the display's common cathode pin to the 0V rail and activating any segments with 5V on their anode terminals.

If we wished to perform similar experiments with a common-anode display, we would use a PNP transistor as illustrated in Fig.7(d). In this case, we might opt for a BC327 (there are, of course, many other candidates for the job). This works in a complementary fashion to its NPN counterpart.

In this case, a 0V signal applied to the transistor's base resistor will switch it on, thereby connecting the display's common anode pin to the 5V rail and activating any segments with 0V on their cathode terminals. By comparison, a 5V signal applied to the transistor's base will switch it off, thereby disconnecting the display's common anode pin from the 5V rail and ensuring all the display segments are off.

Want an easy way to remember which symbol is which? Refer to the direction of the arrow at the transistor's emitter terminal:

NPN = Not Pointing iN
PNP = Point iN Please

DIY or off the shelf?

Finally, we are ready to add a BCD to 7-segment decoder to our circuit (Fig.8). In this case, we are going to employ the simplest form of BCD code, known as Natural BCD (NBCD), Simple BCD (SBCD) or 8421 BCD (see last month's column for more on this).

We've changed from using uppercase letters A, B, C, D, E, F, G to lowercase a, b, c, d, e, f, g to refer to the segments on our 7-segment display. This is because we are now using the uppercase A, B, C, D characters to refer to the inputs to our BCD decoder. It's not my fault; I'll explain more in a moment.

When it comes to implementing our BCD decoder, we have a choice: DIY (do it yourself) or off the shelf. By DIY, I mean that, similar to how we realized our 2:4 decoder, we could use a bunch of primitive logic gates (NOT, AND, OR, etc) to implement it. We'd start by thinking of the decoder as a black box, then create a truth table showing the gazins and gazouts. Next, we'd use techniques like Boolean algebra, De Morgan transformations and Karnaugh maps to determine which logic gates to use.

Once again, all of this is discussed in *Bebop to the Boolean Boogie*. The

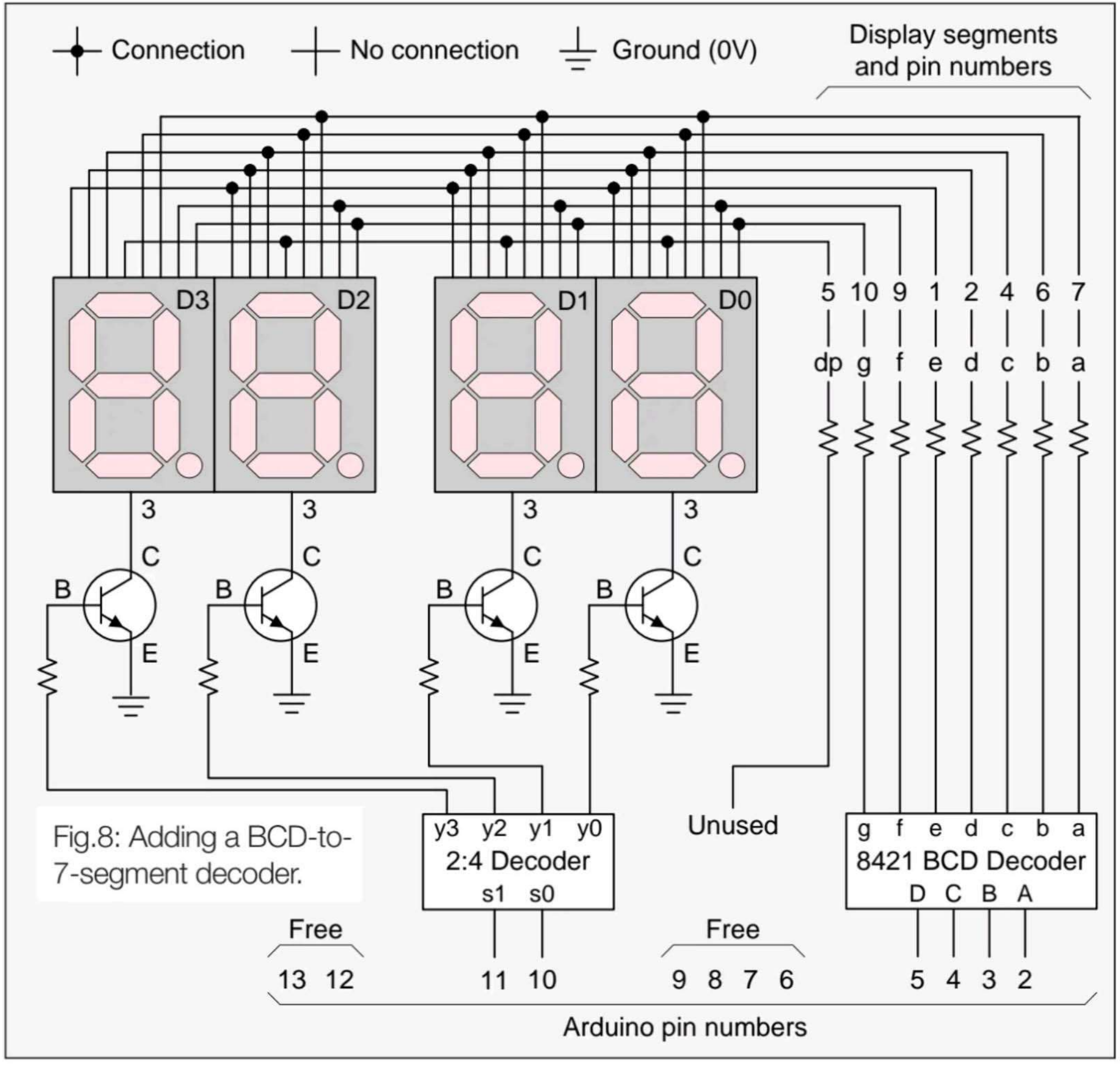


Fig.8: Adding a BCD-to-7-segment decoder.

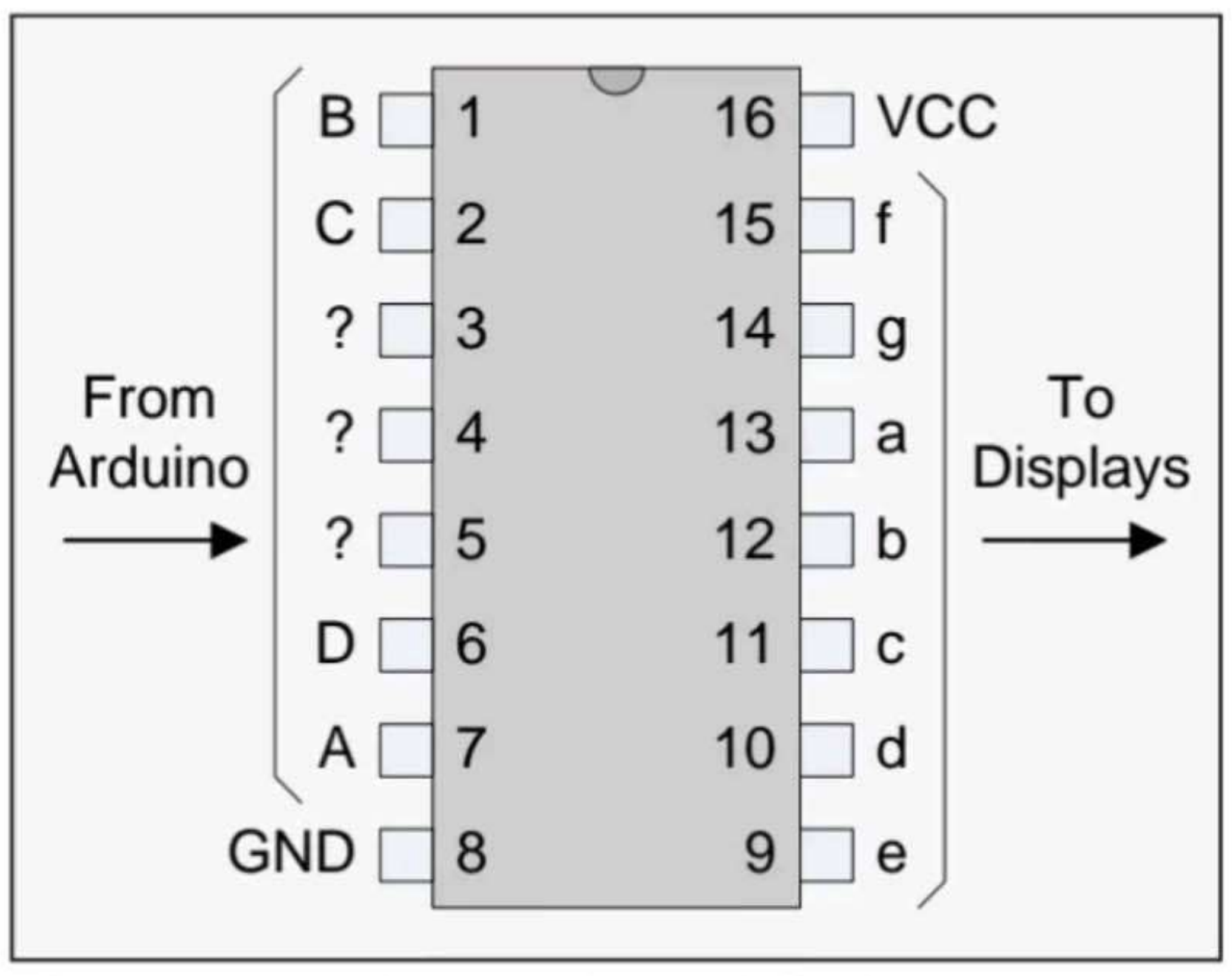


Fig.9: A top view of the 74LS48N package.

problem is that this would take time and effort and I'm feeling lazy today. The alternative is to use an off-the-shelf part like the SN74LS48N. I just found a pack of 10 for £5 on Amazon's UK website (**https://pemag.au/link/abzj**), but you can get these little scamps from any major component supplier.

As we discussed in an earlier column (*PE*, August 2024), the "SN" indicates that Texas Instruments (TI) is the manufacturer, the "74" tells us we are working with a commercial component (as opposed to industrial or military), the "LS" informs us that this is a member of the Low-Power Schottky family, the "48" is the type of logic function (a BCD decoder), and the "N" indicates it is in a plastic dual in-line package (PDIP).

Data sheet deliriums

It's when we come to cast our orbs over the data sheet for the 74LS48 that things have the potential to get a little confusing (**https://pemag.au/link/abzk**). Whoever was charged with creating the original version of this little rascal back in the 1970s was overly enthusiastic because they made it their mission to cover a wide variety of parts, packages and functions in this one document.

We are focusing on the 74LS48 because this device is intended to drive a common-cathode 7-segment display, which means its outputs are active-high. Its 74LS47 counterpart, also embraced by the same data sheet, has active-low outputs and is designed to drive a common-anode 7-segment display.

Based on this data sheet, a representation of the 16-pin package we're using is presented in Fig.9. This explains why we transitioned to using A, B, C, D characters for the inputs to the decoder and a, b, c, d, e, f, g characters for the signals driving our 7-segment displays.

Decimal	Inputs				Outputs						
	D	C	B	A	a	b	c	d	e	f	g
0	L	L	L	L	H	H	H	H	H	H	L
1	L	L	L	H	L	H	H	L	L	L	L
2	L	L	H	L	H	H	L	H	H	L	H
3	L	L	H	H	H	H	H	H	L	L	H
4	L	H	L	L	L	H	H	L	L	H	H
5	L	H	L	H	H	L	H	H	L	H	H
6	L	H	H	L	L	L	H	H	H	H	H
7	L	H	H	H	H	H	H	L	L	L	L
8	H	L	L	L	H	H	H	H	H	H	H
9	H	L	L	H	H	H	H	L	L	H	H
10	H	L	H	L	L	L	L	H	H	L	H
11	H	L	H	H	L	L	H	H	L	L	H
12	H	H	L	L	L	H	L	L	L	H	H
13	H	H	L	H	H	L	L	H	L	H	H
14	H	H	H	L	L	L	L	H	H	H	H
15	H	H	H	H	L	L	L	L	L	L	L

Fig.10: The 74LS48 truth table.

The GND pin (pin 8) connects to the 0V ground rail on our breadboard, while the VCC power pin (pin 16) will connect to the +5V power rail. The "?" for pins 3, 4, and 5 is because we will consider the functioning of these pins next month. For the moment, we simply use a 10kΩ resistor to disable these pins by pulling them up to logic 1 (the 5V rail).

A cut-down version of the truth table from Table T2 in the data sheet is shown in Fig.10 (we've omitted pins 3, 4, and 5).

In our previous columns, we've predominantly used 0s and 1s in our truth tables (along with occasional "X" characters to indicate 'Don't Care' values on the inputs and "?" characters to indicate 'Don't Know' values on the outputs).

In the past, it was not uncommon to use "L" and "H" instead of 0 and 1, respectively. I'm continuing to use "L" and "H" here to match the data sheet and because you might run across this sort of thing in the future.

Although this isn't a concern here, it's worth noting that some truth tables might have "OFF" and "ON" values for their outputs. Oftentimes, this is because the creators of the data sheet were trying to cover multiple device variants, some with active-high outputs and others with active-low outputs. Using "OFF" and "ON" values allows one truth table to cover both implementations.

One final point of interest from our truth table arises when we consider the numerical designations and resultant displays illustrated in Fig.11.

The numbers 0 through 9 are displayed as we would expect, but what about 10 through 15? We introduced hexadecimal (base 16) in *PE*, May 2023. Did the designers of this device miss their opportunity to come up with a hexadecimal display? Well, yes and no.

On the one hand, many people in the computer industry used only octal (base 8) and decimal (base 10) notations into the late 1970s and early 1980s, and the octal characters (0 to 7) form a subset of the decimal characters (0 to 9). On the other hand, starting in the early 1960s, it became increasingly common to use hexadecimal (base 16) notation. So displaying the hex A, B, C, D, E, F characters would almost certainly have been considered by its designers.

The telling point here is the fact that number 15 corresponds to all the segments being off. That is useful if we wish to blank leading zeros on a multi-digit display. We introduced this concept in *PE*, August 2024, and we will be discussing it in more detail next month.

That leaves us with the displays corresponding to numbers 10 through 14. Were these cryptic segment patterns carefully selected by the device's designers? Do they have some long-forgotten meanings? Personally, I doubt it. I guess that the designers focused on getting the displays they desired for numbers 0 through 9, along with the blank (all segments off) corresponding to number 15, leaving the displays corresponding to numbers 10 through 14 to fall out as they might (I'd love to hear if you have another theory or know something to the contrary).

Preparing to add the BCD decoder

Before we add the 74LS48 BCD decoder to our breadboard, let's remind ourselves as to the current state of play. As usual, you can download a PDF of our latest and greatest breadboard, component, and wiring ensemble (file "CB-Oct24-01.pdf"). All the files mentioned in this column are available from the October 2024 page of the PE website: **https://pemag.au/link/abzl**

To make sure everything is still working before making any changes, let's load and run our most recent program, which displays the time while suppressing any leading 0 in the most-significant (left-

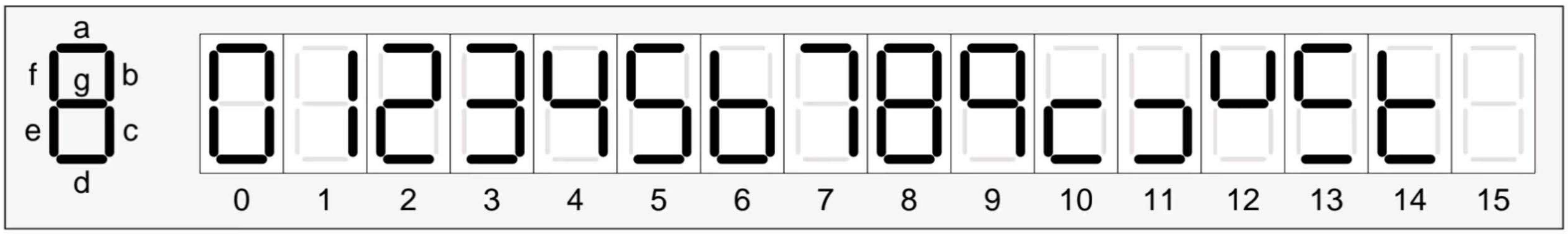


Fig.11: Segment identification, numerical designations and resultant displays from the 74LS48 data sheet.

most) digit (file "CB-Oct24-02.txt"). If everything is working as expected, we can proceed.

The area of interest to us with respect to our existing implementation sans BCD decoder is illustrated in Fig.12(a). As you will recall, since we are multiplexing our displays, this allows us to connect the LED segments from all four displays in parallel (side-by-side). We're using digital output pins 2 through 9 from the Arduino to drive display segments 'a' through 'f' and the decimal point ('dp'), respectively. We are currently using eight 150Ω discrete (individual) current-limiting resistors mounted on a small breakout board (BOB). We discussed its creation in *PE*, July 2024.

Now we will swap out the current-limiting resistors with new values and add the BCD decoder as illustrated in Fig.12(b). From previous columns we know the red LEDs in our 7-segment displays have a forward voltage drop (V_F) of 2V and a maximum forward current (I_F) of 20mA (we discussed the concept of a diode's forward voltage drop in *PE*, March 2024).

The data sheet for the 74LS48 states that the maximum current each output can provide is only 6mA, so that is what we have to work with.

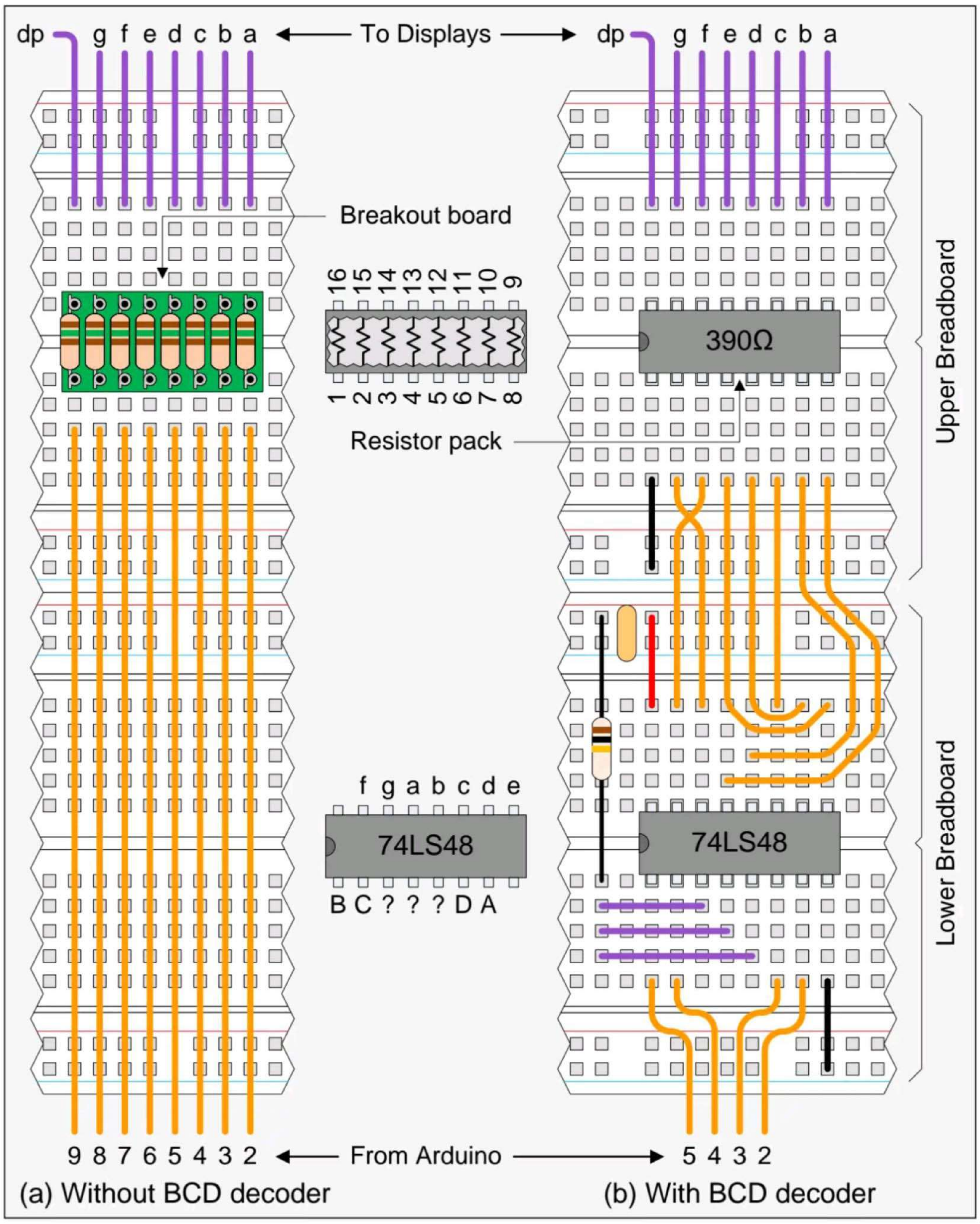
Remember that we have NPN transistors in series with each of our displays. These have a 'saturation voltage' that appears across them when they are switched on, usually a fraction of a volt. This means that we end up with only about 5V – 2V – 0.5V = 2.5V driving each LED. Using Ohm's law of V = I × R and rearranging the terms to give R = V ÷ I, we end up with R = 2.5V ÷ 0.006A = 416Ω. The closest standard value to this is 390Ω, which is close enough.

You can either build a new BOB or do what I did, which is to use a 16-pin package containing eight isolated 390Ω resistors. These components are created by a variety of manufacturers, like the Bourns 4100R Series (https://pemag.au/link/abzm), and they are available from most major component vendors, like DigiKey (https://pemag.au/link/abzn).

Since we can't control the decimal point ('dp') segments on our 7-segment displays using our BCD decoder, we will connect pin 1 of the resistor pack to ground, thereby disabling these segments. Also, we've used a 10kΩ pull-up resistor to ensure that pins 3, 4, and 5 on the 74LS48 are in their inactive states (we will consider the functioning of these pins next month).

Bringing-up the BCD decoder

Based on our previous experiences of things not always working as planned the first time around, we will take things step by step.



(a) Without BCD decoder (b) With BCD decoder

Fig.12: Adding the BCD decoder chip to our breadboard.

First, unplug the USB cable from your Arduino to power it down. While working on an antistatic mat and wearing an antistatic wrist strap, disconnect the breadboard ends of the wires connected to pins 2, 3, 4, and 5 on the Arduino. Also, completely remove the wires connected to pins 6, 7, 8, and 9 on the Arduino. Now replace the original 150Ω BOB with your new 390Ω BOB or resistor pack on the upper breadboard (remember to connect pin 1 on the resistor pack/BOB to ground).

Next, add the SN74LS48 IC to the lower breadboard, as in Fig.13(a). Also add a black wire from pin 8 (GND) to the lower ground rail, followed by a red wire from pin 16 (V_{CC}) to the upper +5V power rail. Also, add the 0.1μF (100nF) ceramic capacitor and the 10kΩ pull-up resistor, along with the three purple wires connected to pins 3, 4, and 5.

Now reconnect the USB cable from

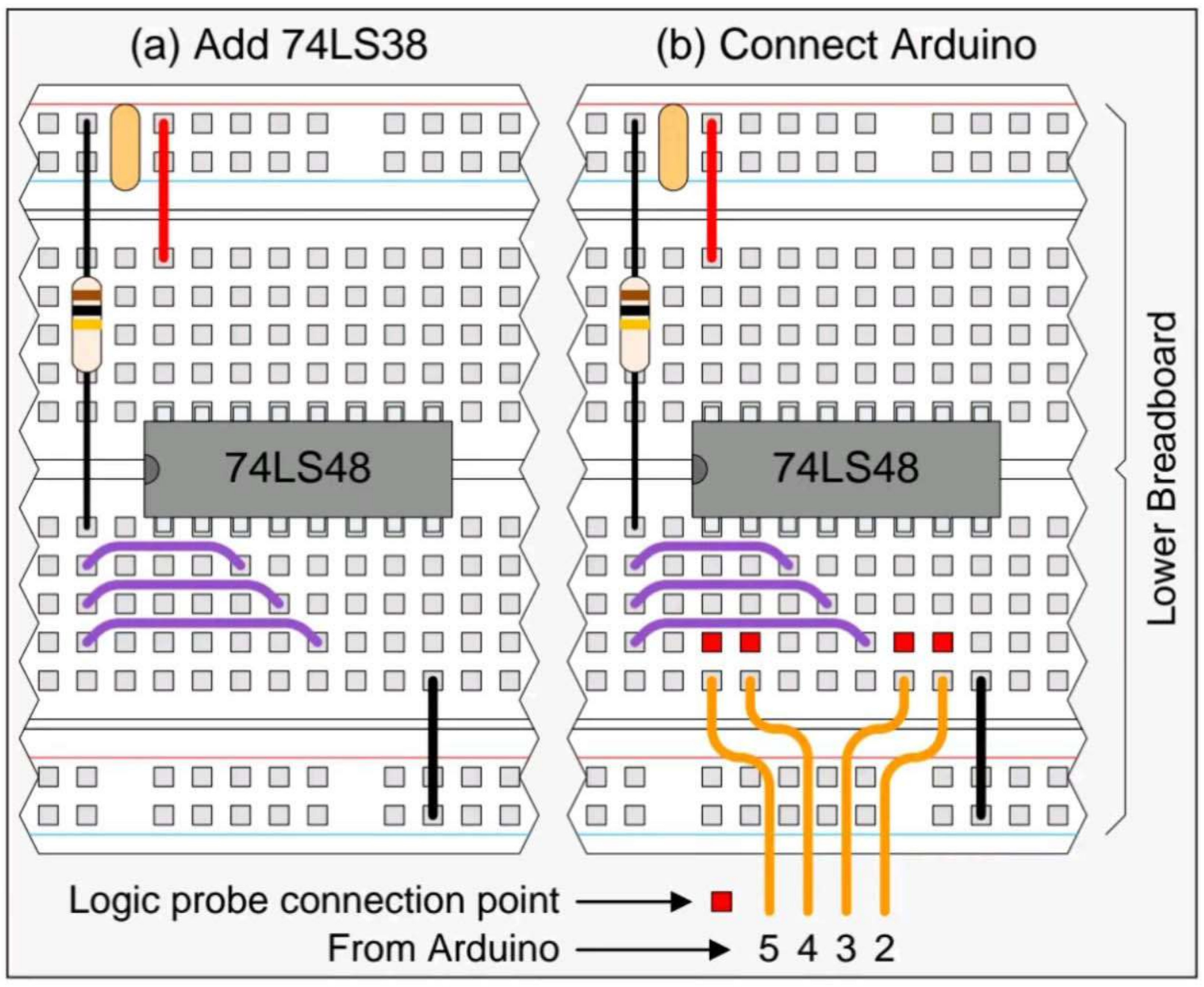


Fig.13: Upgrading our breadboard one step at a time.

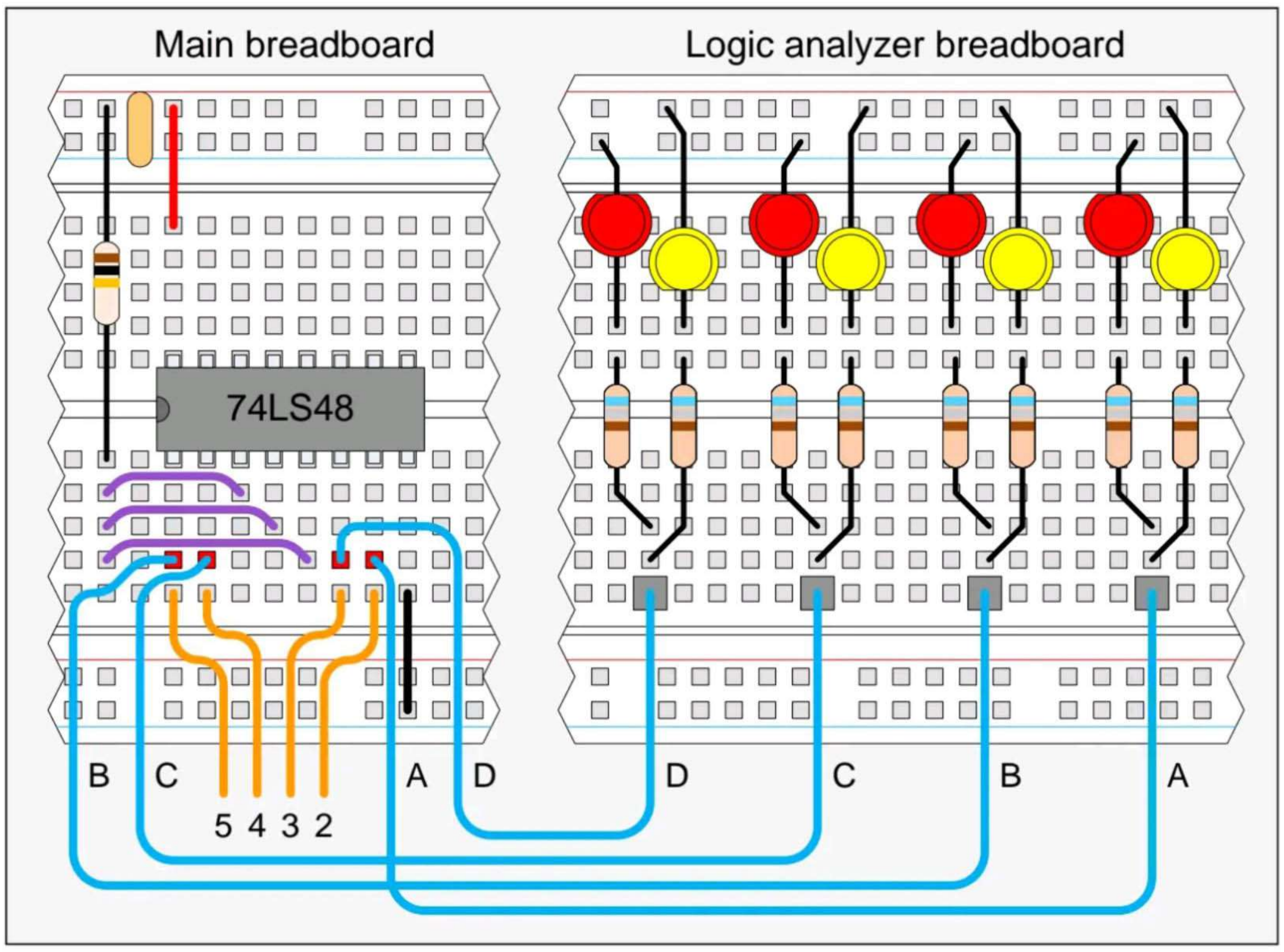


Fig.14: Connecting our LED-based logic analyser to the 74LS48 IC.

your host computer to your Arduino. Don't worry if your 7-segment displays are glowing dimly or showing spurious values. This is because we have not yet connected the outputs from the SN74LS48 to the 'inputs' of the current-limiting resistors.

As we previously discussed (*PE*, August 2024), use your digital multimeter to ensure we have 5V (or thereabouts) between pin 8 and pin 16 on the IC (apply the multimeter's probes directly to the IC's pins). Next, leave the black probe on pin 8 and use the red probe to ensure we see the same 5V (give-or-take) value on pins 3, 4, and 5.

So far, so good. Remove the USB cable to power everything down again, then reconnect the breadboard ends of the wires from pins 2, 3, 4, and 5 on the Arduino as illustrated in Fig.13(b). Observe that, the way we've wired things, inputs A, B, C, D on the 74LS48 (pins 7, 1, 2 and 6, respectively) are connected to pins 2, 5, 4, 3 on the Arduino, respectively.

Don't panic. I know we haven't connected the outputs from the BCD decoder yet. For the moment, we are going to focus on its inputs. Wrapping one's brain around all of this (which pins correspond to which signals and wires) can require some mental gymnastics. So, before we do anything else, we need to make sure we know what we're doing and that we are controlling things the way we want to control them.

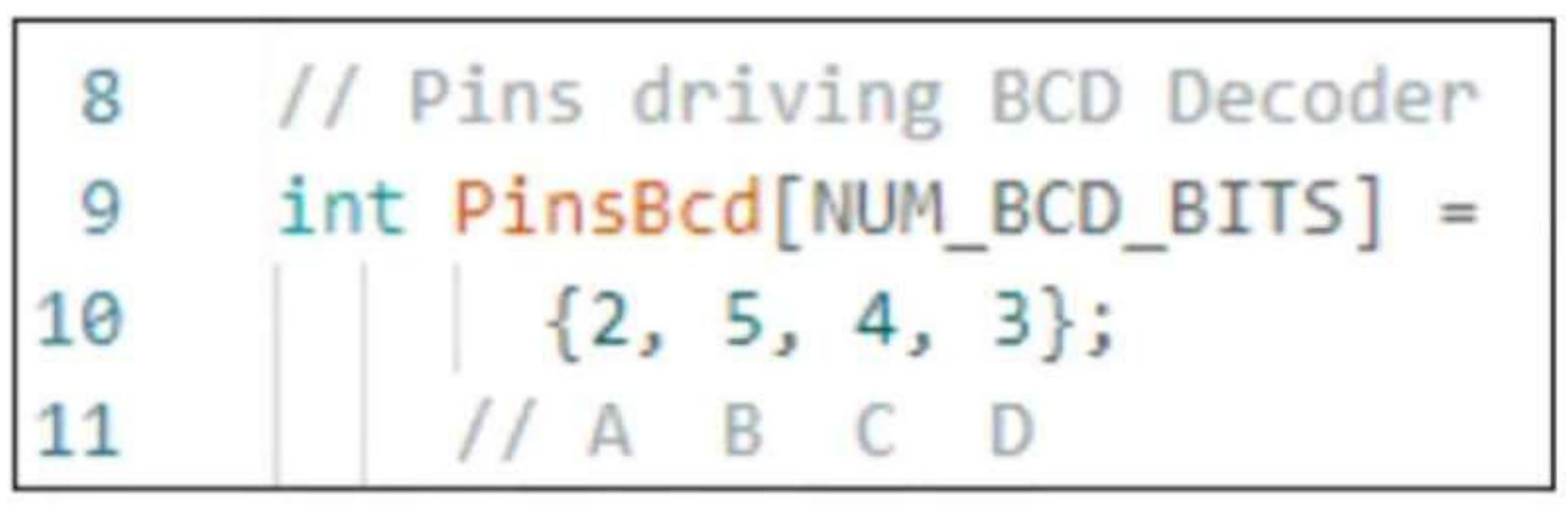
To help with this, we're going to use four of the probes on the logic analyser board we constructed earlier (*PE*, August 2024), as illustrated in Fig.14. Don't forget to connect the logic analyser's upper power and ground rails to their counterparts on the main breadboard.

Before we power everything up, let's first create a small test program to make sure we are talking to the BCD decoder in a language it understands (file "CB-Oct24-03.txt"). As usual, we won't go through everything here – we'll just hit the highlights.

As you'll see when you look at the code, we are using variables of type int (integer) to represent our BCD values. Integers are stored as 16-bit values in the Arduino Uno, but we will be using only the four least-significant bits to represent values 0 through 9 in decimal (0000 through 1001 in binary). As part of this, we define `NUM_BCD_BITS` to be 4.

We declare a global integer variable called `ValBcd` (think "BCD value"). We also declare an array of integers called `PinsBcd[]` to store the numbers of the Arduino pins we wish to use to drive our BCD decoder. This declaration is illustrated in Listing 3a.

In the `setup()` function, we specify that the pins driving our 2:4 decoder are outputs and we set them both to `LOW`, thereby selecting display D0 (which we will be using a little later). Next, we specify that the pins driving our BCD decoder are outputs and set them `LOW` also. Last, but not least, we assign a value of 0 to our `ValBcd` variable.



```
// Pins driving BCD Decoder
int PinsBcd[NUM_BCD_BITS] =
      {2, 5, 4, 3};
    // A  B  C  D
```

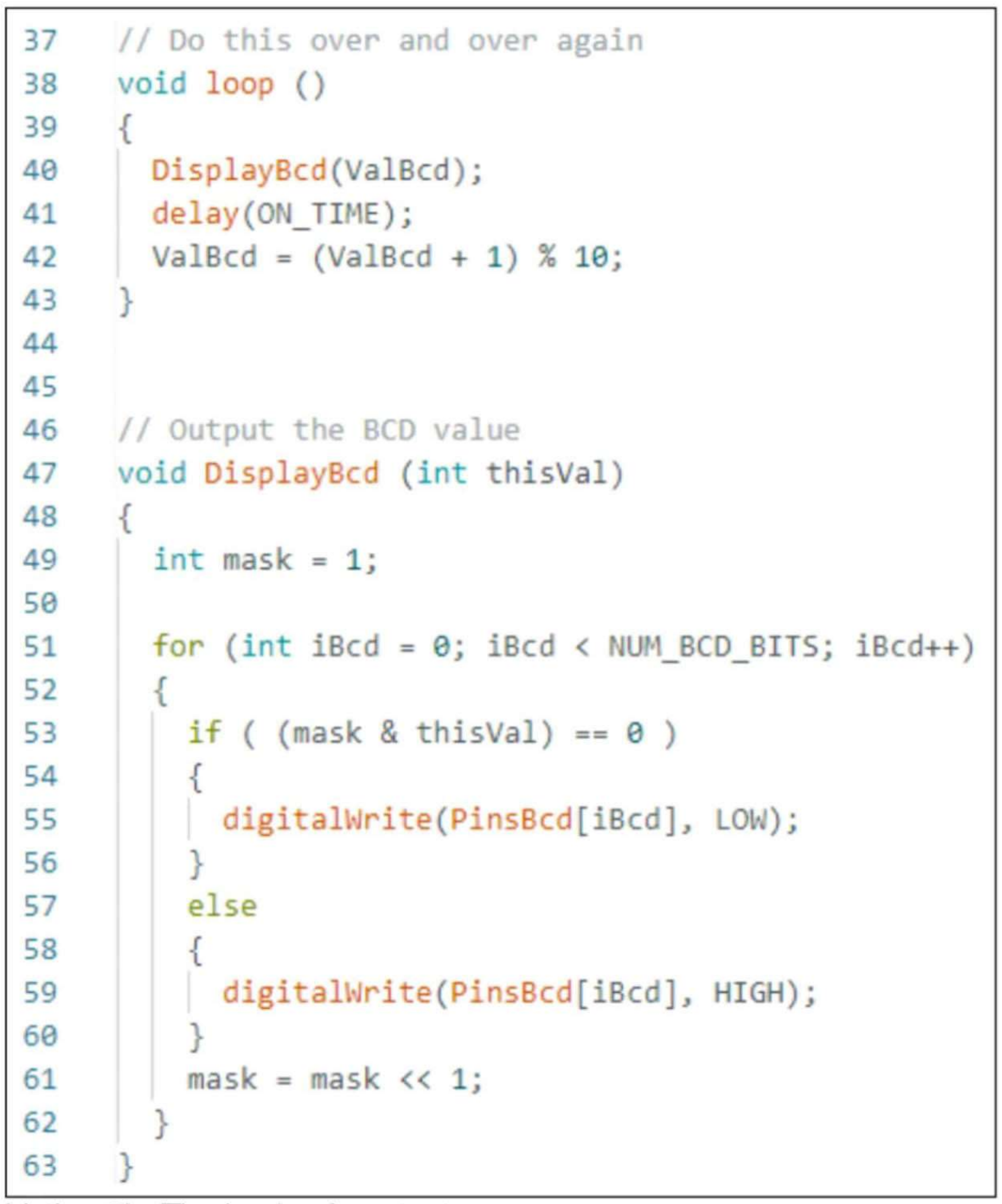
Listing 3a: Our pin declarations for testing.

The main part of our test program is shown in Listing 3b. Let's start with the new function called `DisplayBcd()`, into which we pass the BCD value we wish to display. This works similarly to the `DisplaySegs()` function we've been using to drive our displays directly. We described this in detail in *PE*, April 2023. All we need to note here is that our new function will present bits 0, 1, 2, and 3 of our BCD value to Arduino pins 2, 5, 4, and 3, respectively.

Finally, we have the `loop()` function, which is as simple as simple can be. It starts on Line 40 by calling our new `DisplayBcd()` function to display the current value stored in `ValBcd`.

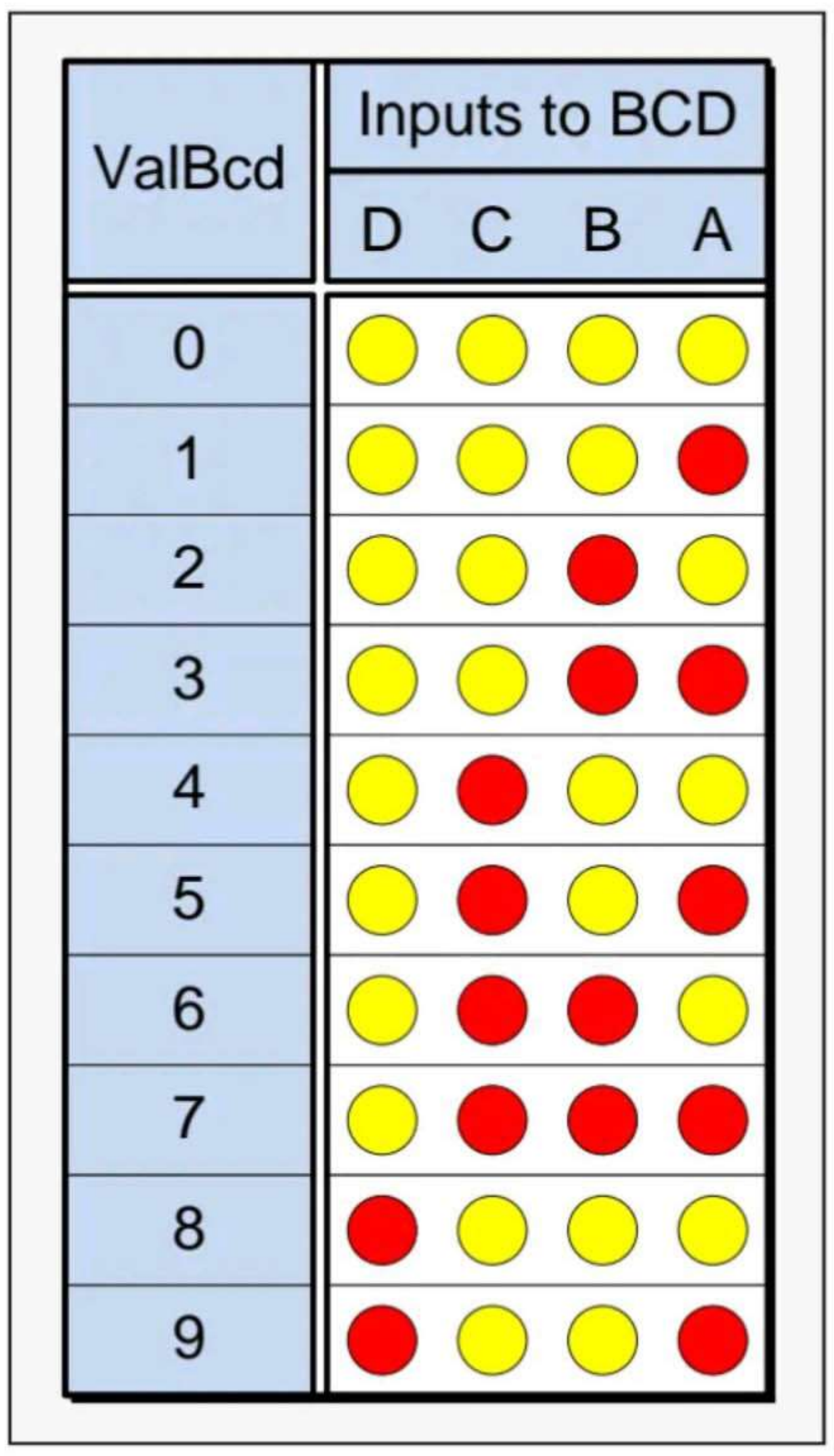
On Line 41, we pause for `ON_TIME`, which we've set to one second. On Line 42, we increment the value stored in `ValBcd`. The use of the modulus/integer remainder (%) operator means our count will follow the sequence 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 and back to 0 again.

OK, let's power up our Arduino and then load and run our test program. Changing each second, our logic analyser LEDs should reflect the binary



```
37  // Do this over and over again
38  void loop ()
39  {
40    DisplayBcd(ValBcd);
41    delay(ON_TIME);
42    ValBcd = (ValBcd + 1) % 10;
43  }
44
45
46  // Output the BCD value
47  void DisplayBcd (int thisVal)
48  {
49    int mask = 1;
50
51    for (int iBcd = 0; iBcd < NUM_BCD_BITS; iBcd++)
52    {
53      if ( (mask & thisVal) == 0 )
54      {
55        digitalWrite(PinsBcd[iBcd], LOW);
56      }
57      else
58      {
59        digitalWrite(PinsBcd[iBcd], HIGH);
60      }
61      mask = mask << 1;
62    }
63  }
```

Listing 3b: The body of our test program.



ValBcd	Inputs to BCD			
	D	C	B	A
0	🟡	🟡	🟡	🟡
1	🟡	🟡	🟡	🔴
2	🟡	🟡	🔴	🟡
3	🟡	🟡	🔴	🔴
4	🟡	🔴	🟡	🟡
5	🟡	🔴	🟡	🔴
6	🟡	🔴	🔴	🟡
7	🟡	🔴	🔴	🔴
8	🔴	🟡	🟡	🟡
9	🔴	🟡	🟡	🔴

Fig.15: The expected logic analyser display.

counting sequence depicted in Fig.15. If not, return to the instructions above and recheck the wiring.

Once you have this working, power everything down, then add the wires connecting the outputs from the BCD decoder to the inputs of the current-limiting resistors, as illustrated in Fig.12(b).

As usual, you can download a PDF showing our new BCD-based breadboard setup in its entirety (file "CB-Oct24-04.pdf"). When you return power to the Arduino, you should see the values 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0… appear on display D0.

All together now!

Finally, it's time to bring everything together. We're going to take our old non-BCD clock program (file "CB-Oct24-02.txt") and the BCD test program we just created (file "CB-Oct24-03.txt"), and "munge" them together to create a new BCD-based clock program.

Why don't you do this yourself to make sure you are totally on top of things? If you run into any problems, you can compare your solution to mine (file "CB-Oct24-05.txt").

OMG, it works! (I mean, I never doubted us for a minute...)

You may notice that the 7-segment displays do not appear as bright as they used to be. There's a reason for this, which is that they're not as bright as they used to be. Previously, we were using 150Ω current-limiting resistors to drive the displays with 20mA. Since we were multiplexing four displays, each display was on for only 25% of the time. That meant each display used to receive 20mA for 25% of the time and 0mA for 75% of the time, averaging out at 5mA per display.

Components from Part 1

LEDs (assorted colours)	https://amzn.to/3E7VAQE
Resistors (assorted values)	https://amzn.to/3O4RvBt
Solderless breadboard	https://amzn.to/3O2L3e8
Multicore jumper wires (male-to-male)	https://amzn.to/3O4hnxk

Components from Part 2

7-segment display(s)	https://amzn.to/3Afm8yu

Components from Part 5

Momentary pushbutton switches	https://amzn.to/3Tk7Q87

Components from Part 6

Passive piezoelectric buzzer	https://amzn.to/3KmxjcX

Components for Part 9

SW-18010P vibration switch	https://bit.ly/46SfDA4

Components for Part 10

Breadboard mounting trimpots	https://bit.ly/3QAuz04

Components for Part 12

Light-Dependent Resistor	https://bit.ly/3S2430m

Components for Part 13

BC337 NPN Transistor	https://bit.ly/40xAgyS

Components for Part 14

HC-SR04 Ultrasonic Sensor	https://bit.ly/49AMBq4

Components for Part 15

Real-Time Clock (RTC)	https://bit.ly/3S9OjHl

Components for Part 18

Long tailed (0.1-inch/2.54mm pitch) header pins	https://bit.ly/3U1Vp2z

Components for Part 19

Prototyping boards	https://bit.ly/3UMkcZ1
Kit of popular SN74LS00 chips	https://bit.ly/3wqgzyv

Components for Part 20

16V 100µF electrolytic capacitors	https://bit.ly/44LzpNa
Ceramic capacitors (assorted values)	https://bit.ly/4bEAUiv

Components for Part 22

SN74LS48N BCD Decoders	https://bit.ly/3zT18jx
16-pin resistor pack (8 × 390Ω)	https://bit.ly/4d0ISDz

Now, due to our 74LS48 BCD decoder's outputs being limited to only 6mA, we've changed our current-limiting resistors to 390Ω, so each display is seeing an average of only 1.5mA, which explains why they appear dimmer now.

Blowing one's own flügelhorn

Modesty forbids me from being boastful. Happily, members of the *PE* community are usually pleased to oblige.

For example, I just received an email from a reader we will call Ian (because that's his name). He wrote:

Dear Max, I have to get this boast in while acknowledging that I've obviously learned lots from you over the last year and a half. Some time ago, I bought a set of 4-digit, 7-segment displays from AliExpress.

The problem was the unit is common anode. I am quite proud of myself as I was able to work out which transistors to use. Also, which resistors for both the LEDs and the transistor base. I'm happy to report that it worked first time (that's a first)!

It was the "I've obviously learned lots from you over the last year and a half" portion of this message that caught my eye. My mother will be so proud. We always suspected that someone other than her was reading these columns. Now we know his name!

Next time

In our next column, we will dive deeper into the functioning of the 74LS48 decoder.

After that, we will introduce a different BCD decoder integrated circuit that will better serve our needs, including addressing our lack-of-brightness issue.

Until that frabjous day, if you have any thoughts you'd care to share, as always, I welcome your insightful comments, penetrating questions, and sagacious suggestions. PE

Linshang LS172 Colorimeter

If you want to find a paint colour that matches your existing paint or verify that batches of products you are ordering have matching colours, this device is perfect. It's relatively inexpensive, very accurate and quite easy to use. It even costs less than a single colour sensor we recently looked at!

Review by Allan Linton-Smith

Background image: https://unsplash.com/photos/46juD4zY1XA

The LS172 is a hand-held colorimeter that's ready to use out of the box. It is capable of accurate measurements and allows you to easily compare the colours of different objects. It uses reflected light from a LED beamed onto a sample, analysing colour and intensity.

It is an essential tool for anyone involved with colour measurement, such as painters, decorators, or anyone requiring standardisation of coloured items. It gives you CIELAB (L*a*b*) measurements (when translated from French to English, CIE stands for the International Commission on Illumination) and can also measure tiny colour differences. Additionally, it can convert colour values into Pantone numbers, which printers and colour consultants often refer to.

It costs around $250-300 depending on where you buy it; it is available from eBay, Amazon, AliExpress and other online retailers. That might sound expensive, but it's cheap compared to what was available before! For comparison, the Omron BW5C colour sensor can be purchased from DigiKey for around $280. And that's just for the sensor!

The LS172 is undoubtedly good at matching house paint; I recently used it to match my house colour to a colour sample at the hardware store. I used its memory to store my house colour CIELAB measurements. I then checked out similar colour sample cards until it gave a "green" reading, indicating that the sample was almost identical to the house colour.

The colour it identified was "Beige Royal", which it indicates is similar to Pantone 7527 C or Pantone 7434 C. It also tells me that Beige Royal is 'more white, slightly more red and slightly more blue' than my house colour.

The device can store 1000 colour measurements, so you can keep your kitchen, bathrooms and other samples for reference if you want!

The hardware store I went to has a colour-matching system but no CIELAB reference or Pantone numbers, meaning I would have to chip off a bit of paint from my house to match it. I much prefer using the LS172 to damaging the house!

Note that smartphone photos are not good enough to match colour accurately because of lens filtering and light source variations, which can easily shift the colour and intensity and result in an imperfect match.

There's also no guarantee that a phone camera has a wide enough colour gamut to distinguish all colours the human eye can.

The CIELAB colour space

The L*a*b* numbers represent any colour and its brightness with three coordinates. The L value indicates the brightness, with 100 being pure white and 0 pure black.

The other two values, a & b, represent the colour's hue and saturation, ranging from -128 to +128. They are the x & y coordinates on a standard colour chart, shown in Fig.1. Note how the centre is unsaturated (grey) and colour saturation increases as you move towards the circle's circumference.

LS172 Features & Specifications

- Illumination: full-spectrum LED, 45° annular illumination, 0° viewing angle
- Measuring aperture: 8mm, 10° field of view
- Measuring time: about one second
- Colour standards supported: CIELAB, Pantone, Luv, LCh, Yxy, CMYK, RGB, Hex
- Colour difference formulas: ΔE*ab, ΔE*uv, ΔE*94, ΔE*cmc(2:1 or 1:1), ΔE*00
- Standard deviation for ΔE*ab: ≤ 0.03
  (average of 30 white tile measurements three seconds after calibration)
- Dimensions: 86×62.5×158mm, 225g
- Power supply: rechargeable 3.7V 4000mAh Li-ion cell;
  10,000 measurements from full charge
- Display: 480×320 pixel IPS colour LCD screen
- Charging port: Type-C USB
- Operating conditions: 0-45°C, 0-85% relative humidity (no condensation)
- Language support: English, Simplified Chinese

LS172 Colorimeter review

Therefore, the a & b coordinates encode both the shade (hue) and colour intensity (saturation). Adding L (brightness) gives you everything you need to define a colour. The entire colour space approximates the range of human daylight vision.

I measured the yellow lid of a Vegemite jar, which was indicated as L=71.6, a=7.8, b=87.4. Interestingly, using the LS172 to compare the lid to the label, it said they did not match and that the label was “more white, more green and more blue”, even though it looked identical to my eyes!

You can find more details on L*a*b* colours at **https://w.wiki/7GRT**

Using it

The LS172 is really easy to use. All you need to do is place it on a flat item such as a wall, door or colour card you wish to match. You can then store its colour as the “standard” and compare it to various samples later, to find a match or determine exactly how different they are. A typical screen image during use is shown in Screen 1.

It calculates the delta (difference) between your standard and your sample and tells you if you pass or fail with a green or red background to the delta bar. There are various options for calculating the delta estimate; we used the standard “ΔE*ab”. You can also change the delta threshold that determines its sensitivity to differences.

As mentioned above, it gives you the delta figures and human-readable text like “more blue” or “more yellow” etc. This can be handy since, as I wrote, they often look identical to the eye. If you were mixing paint, you could use those hints to add a small amount of extra tint to end up with a spot-on mixture.

The LS172 calibrates instantly using the little tile in its protective cover and requires no external calibration.

It is easily recharged with a USB-C charger and you can just put it in your pocket or handbag (although it’s a bit chunky to carry comfortably in our pockets). Its built-in rechargeable Li-ion battery is claimed to be good for 10,000 measurements with a full charge.

It has a user-friendly 3.5-inch (89mm) diagonal touch screen and can quickly match the closest Pantone colour number with a claimed accuracy above 90%.

One of the excellent practical features is the on-screen retention of the saved colour; the comparison colour is also saved in a split screen for easy reference.

How is it?

In summary, the LS172 is a cost-effective tool for those involved in colour specifications, colour analysis and colour control. It is a versatile instrument with an excellent memory and can quickly be set up for various analyses.

For more information, visit the manufacturer’s website at **pemag.au/link/abpi**

Finally, the Editor had a good question: what if you want to match a metallic finish, like many automotive paints? I guess the answer is that it would require a different type of instrument; after all, no instrument can do everything!

A bit of history

Early colour measurements and colour matching were done with pretty crude devices such as a “colour comparator”, a circular dial of various colours that you put against a sample. When it matched by eye, you noted the colour number on the dial. However, it rarely matched perfectly!

For liquids, we used “standardised nestle” tubes, large test tubes of exact dimensions that we held up to daylight to compare with a sample of a previously standardised product.

These methods relied on subjective evaluations and depended on the light source and human judgement. Tungsten light can easily hide colour differences; the LS172 uses pure white light from its LED source.

Also, not everyone has perfect colour vision. Where I worked, an Ishihara test was given to laboratory staff before any colour decisions could be made. Approximately one in 12 males and one in 200 females are colour-blind.

You can take the test yourself at **www.colormax.org/color-blind-test/**

The origin of CIELAB

Richard Hunter developed colorimeters and the L*a*b* system in the 1950s to quantify exact colour hues and intensities using numerical values determined by reflected light. Hunter’s colorimeters were first used

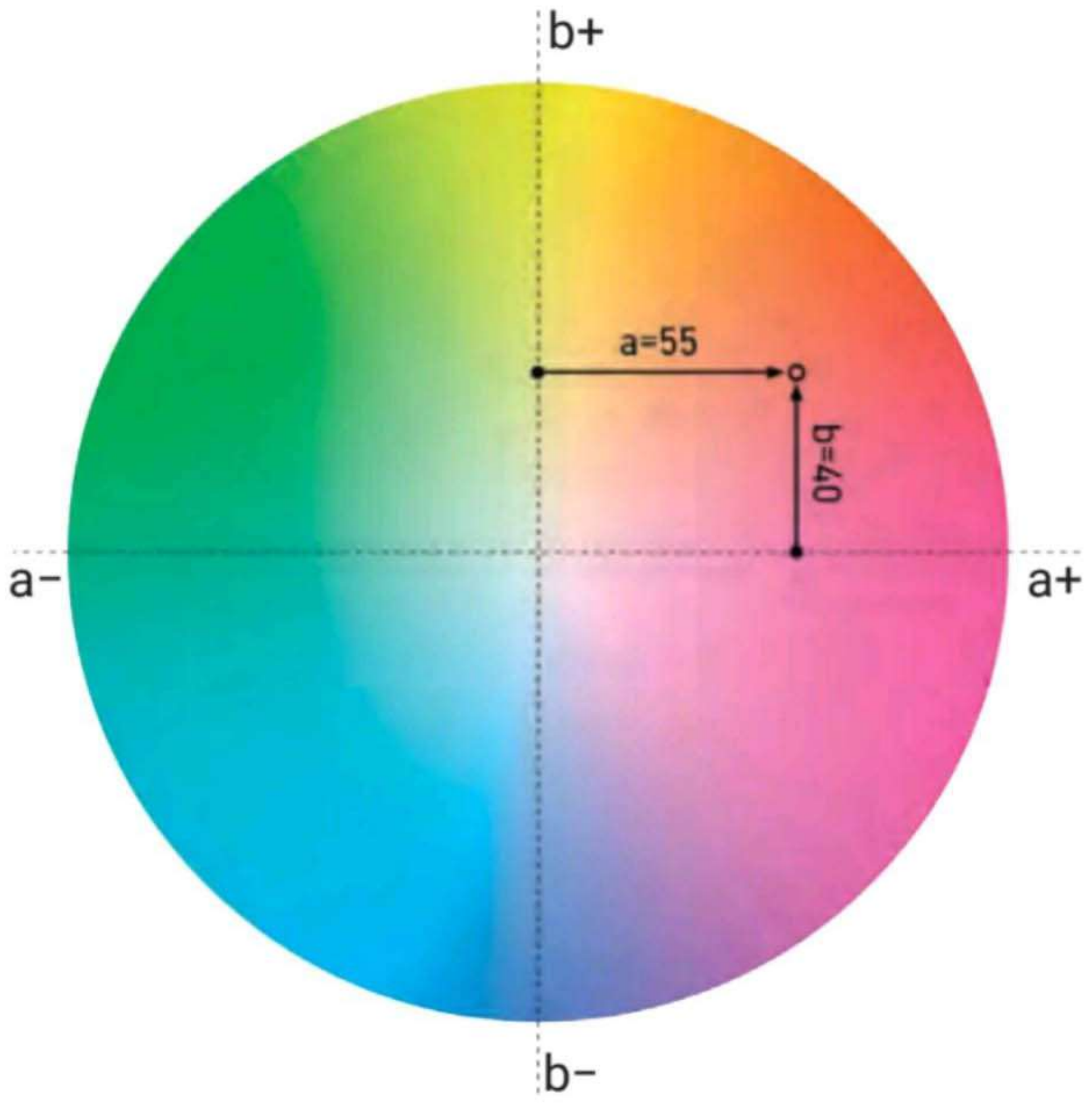
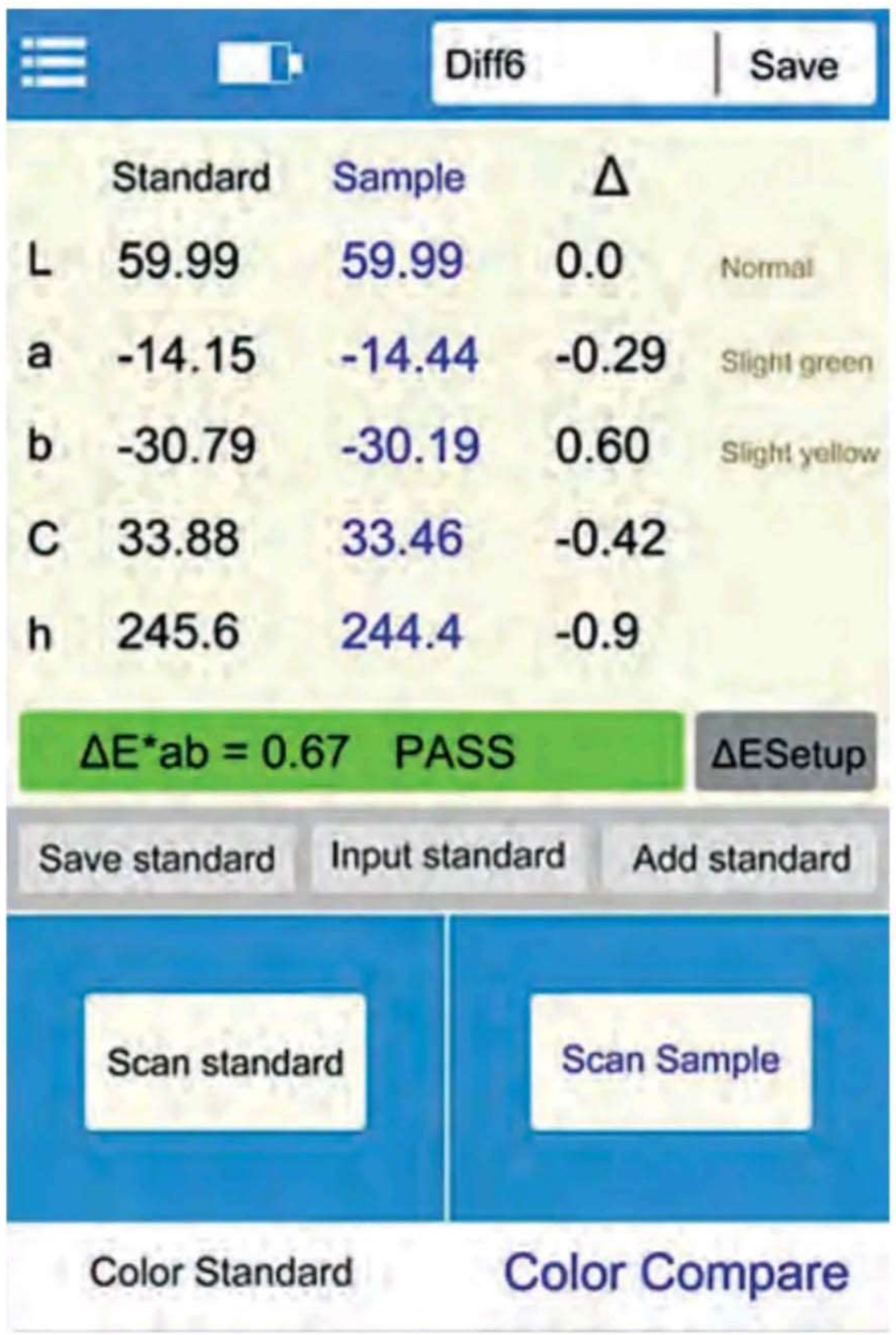


Fig.1: a ‘slice’ through the L*a*b* colour space at around L = 50, halfway between white and black, giving a neutral grey shade in the centre. The example coordinates shown here, a = 55 & b = 40, give a peachy colour. As this is circular, $\sqrt{a^2 + b^2} \leq 128$. Since the magazine is printed in CMYK, this figure will not be displayed accurately. *Source:* https://chromachecker.com/manuals/en/show/chromaspot

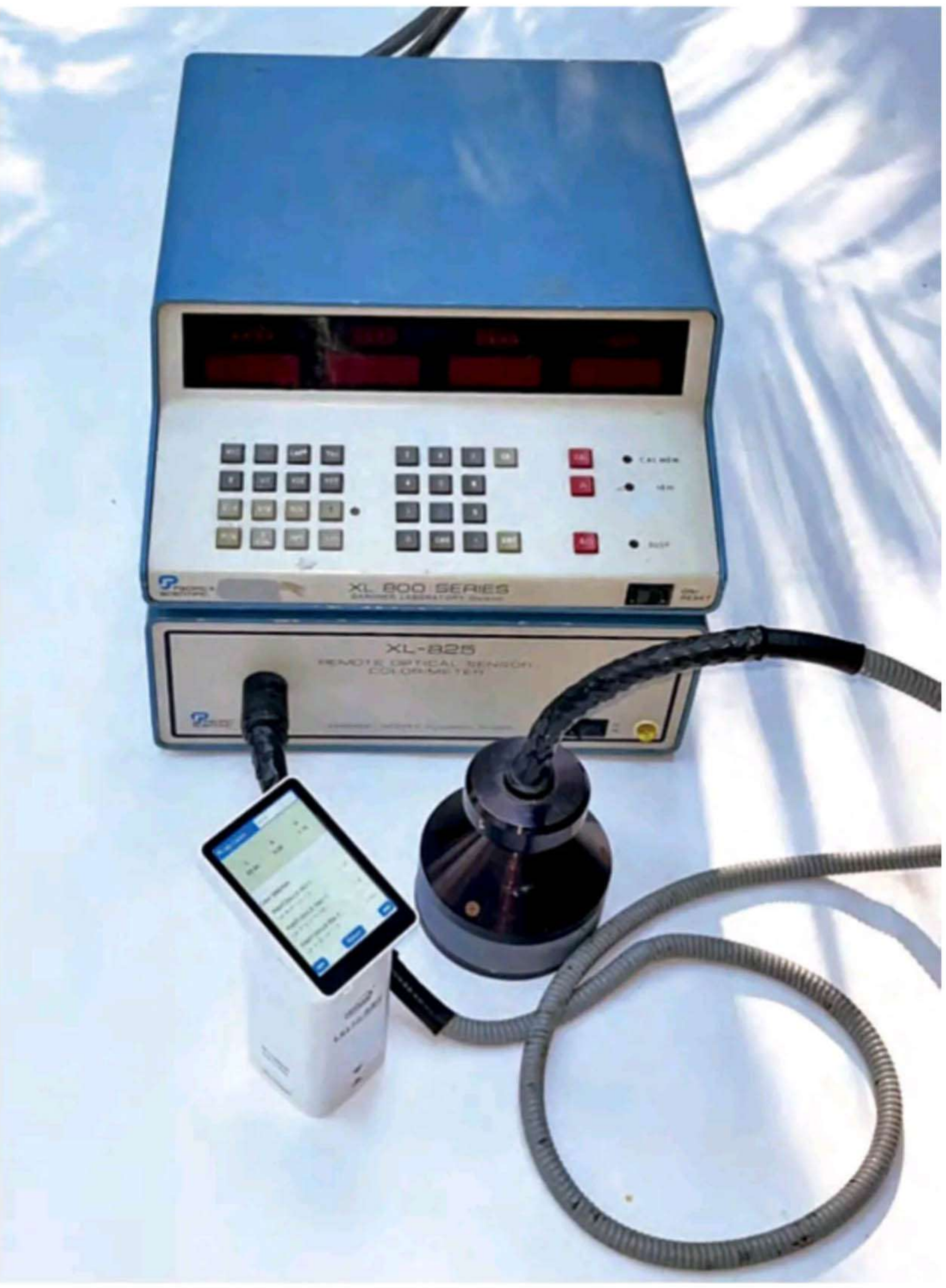


	Standard	Sample	Δ	
L	59.99	59.99	0.0	Normal
a	-14.15	-14.44	-0.29	Slight green
b	-30.79	-30.19	0.60	Slight yellow
C	33.88	33.46	-0.42	
h	245.6	244.4	-0.9	

ΔE*ab = 0.67 PASS

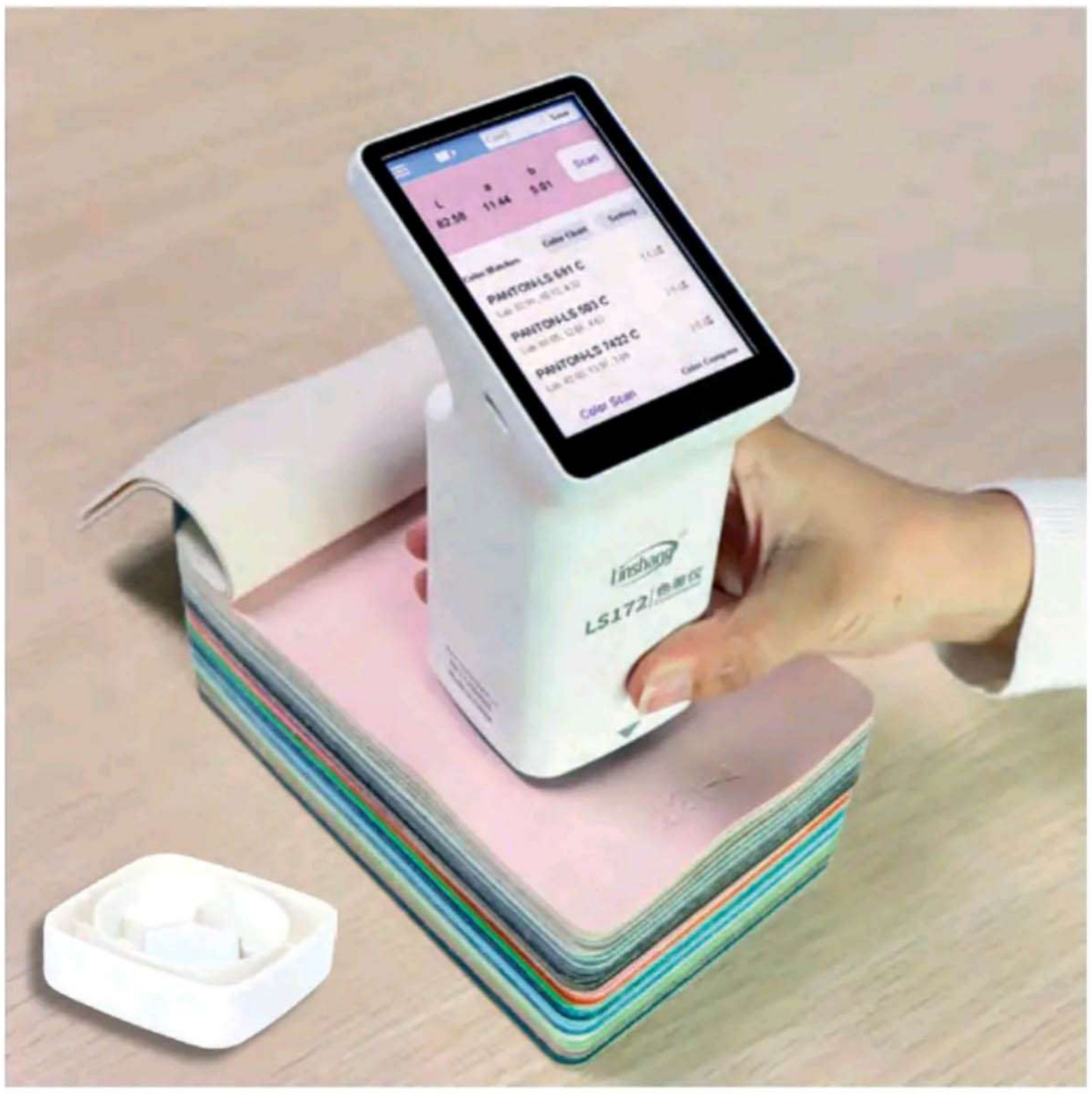
Screen 1: the main screen during use. You can see the measured L, a, b, C & h values at top middle in blue, with the reference values in black to their left. The differences are shown to the right and summarised in the green bar just below the middle of the screen. The perceptual differences are shown on the right. C is for chroma and h is for hue.

LS172 Colorimeter review



Left: a Gardener Laboratory L*a*b* Colorimeter and power supply weighing around 20kg (owned by the author) compared to the LS172. The LS172 is smaller and lighter than the laboratory unit's colour sensor head!

Below: here is the LS172 measuring the colour of a pink sheet. The device provides multiple matches of different Pantone colours. The LS172 also includes a "calibration tile" as part of the bottom cap, which is shown inset. This tile is used as a white reference during calibration.



commercially by Proctor & Gamble to accurately standardise the colours of their soaps. The giant company Dow then adopted them to measure and standardise plastics.

I was fortunate to meet Richard many years ago and was delighted to play around with one of his early colorimeters. Convinced that it would be a game changer in the R&D lab where I worked, I put in a request to get one. I wanted to numerically standardise tomato paste, tomato sauce and various fruit concentrates to ensure consistent quality and help select the best raw materials.

However, the response I got was, "$8,000 for that! Can't you just use your eyes?"

Most serious food labs now have such a colorimeter. The photo above shows my Gardener XL-800 Series display plus XL-825 Optical Remote Sensor Colorimeter. It operates only from 110V AC, consuming over 600W, so it requires a huge 230V to 110V stepdown transformer. It is now obsolete and destined for a museum; a decision I made after testing the little LS172!

Richard Hunter told me that he sold many instruments to forensic labs to compare paint fragments for motor vehicle "hit and run" cases; his instruments were so sensitive that they were able to match paint chips on victims to the exact offending car because no two cars are exactly the same colour! Evidence with L*a*b* measurements is now accepted in most courts.

Spectrometers were also used (and still are) to display colour spectra absorbance of liquids vs wavelength in nanometres over the visible range. However, such measurements require expert interpretation, and you need a large data bank of standard colour spectrograms for comparison.

Early colorimeters such as the Gardener were very heavy, used a lot of power and were consequently restricted to bench work. They were also difficult to calibrate.

Editor's note: while the spelling "colourimeter" is sometimes used (along with "colourimetry"), it is not common. In Australia, it mainly refers to a different type of instrument that measures chemical concentrations by absorption of specific light wavelengths.

Therefore, we are using the spelling 'colorimeter', even though it might appear to be inconsistent with the typical UK/Australian spelling of similar words. PE

What is Pantone?

Pantone is a proprietary colour-matching system developed by the company Pantone LLC. Individual colours are named and matched to a specific printing process (called the Pantone Matching System), with the type and quantity of ink or pigment used cross-referenced to their name.

The importance of Pantone colours comes from the accurate reproduction and standardisation. This means that no matter the location, a specific Pantone colour will display and be printed exactly the same (assuming the printer follows the standard). It's common for logos to be specified using Pantone colour(s), so that no matter if it's printed on a metallic can, wooden box or piece of paper, it will look the same.

DOWNLOAD YOUR PE/EPE BACK ISSUES

COMPLETE-YEAR collections – all 12 issues for each year!

From just £5.95 each

VOL 28 – January to December 1999 – **£5.95**
VOL 29 – January to December 2000 – **£6.95**
VOL 30 – January to December 2001 – **£7.95**
VOL 31 – January to December 2002 – **£7.95**
VOL 32 – January to December 2003 – **£7.95**
VOL 33 – January to December 2004 – **£7.95**
VOL 34 – January to December 2005 – **£8.95**
VOL 35 – January to December 2006 – **£8.95**
VOL 36 – January to December 2007 – **£8.95**
VOL 37 – January to December 2008 – **£8.95**
VOL 38 – January to December 2009 – **£10.95**
VOL 39 – January to December 2010 – **£11.95**
VOL 40 – January to December 2011 – **£12.95**
VOL 41 – January to December 2012 – **£14.95**
VOL 42 – January to December 2013 – **£15.95**
VOL 43 – January to December 2014 – **£16.95**
VOL 44 – January to December 2015 – **£17.95**
VOL 45 – January to December 2016 – **£19.95**
VOL 46 – January to December 2017 – **£22.95**
VOL 47 – January to December 2018 – **£27.95**
VOL 48 – January to December 2019 – **£27.95**
VOL 49 – January to December 2020 – **£29.95**
VOL 50 – January to December 2021 – **£32.95**
VOL 51 – January to December 2022 – **£34.95 – NEW!**

FIVE-YEAR collections – 60 issues!

From just £14.95 each

PE2832: Jan 99 to Dec 03 – £14.95
PE2933: Jan 00 to Dec 04 – £14.95
PE3034: Jan 01 to Dec 05 – £14.95
PE3135: Jan 02 to Dec 06 – £14.95
PE3236: Jan 03 to Dec 07 – £14.95
PE3337: Jan 04 to Dec 08 – £14.95
PE3438: Jan 05 to Dec 09 – £19.95
PE3539: Jan 06 to Dec 10 – £25.95
PE3640: Jan 07 to Dec 11 – £25.95
PE3741: Jan 08 to Dec 12 – £25.95
PE3842: Jan 09 to Dec 13 – £33.95
PE3943: Jan 10 to Dec 14 – £33.95
PE4044: Jan 11 to Dec 15 – £33.95
PE4145: Jan 12 to Dec 16 – £35.95
PE4246: Jan 13 to Dec 17 – £37.95
PE4347: Jan 14 to Dec 18 – £38.95
PE4448: Jan 15 to Dec 19 – £39.95
PE4549: Jan 16 to Dec 20 – £42.95
PE4650: Jan 17 to Dec 21 – £44.95 – NEW!

20 YEARS OF PE!

AMAZING BUNDLE OFFER
Every issue from 1999 – 2018

240 magazines for just

£59.95

PE2832: Jan 99 to Dec 03
PLUS!
PE3337: Jan 04 to Dec 08
PLUS!
PE3842: Jan 09 to Dec 13
PLUS!
PE3447: Jan 14 to Dec 18

Do you have the 15-year bundle?

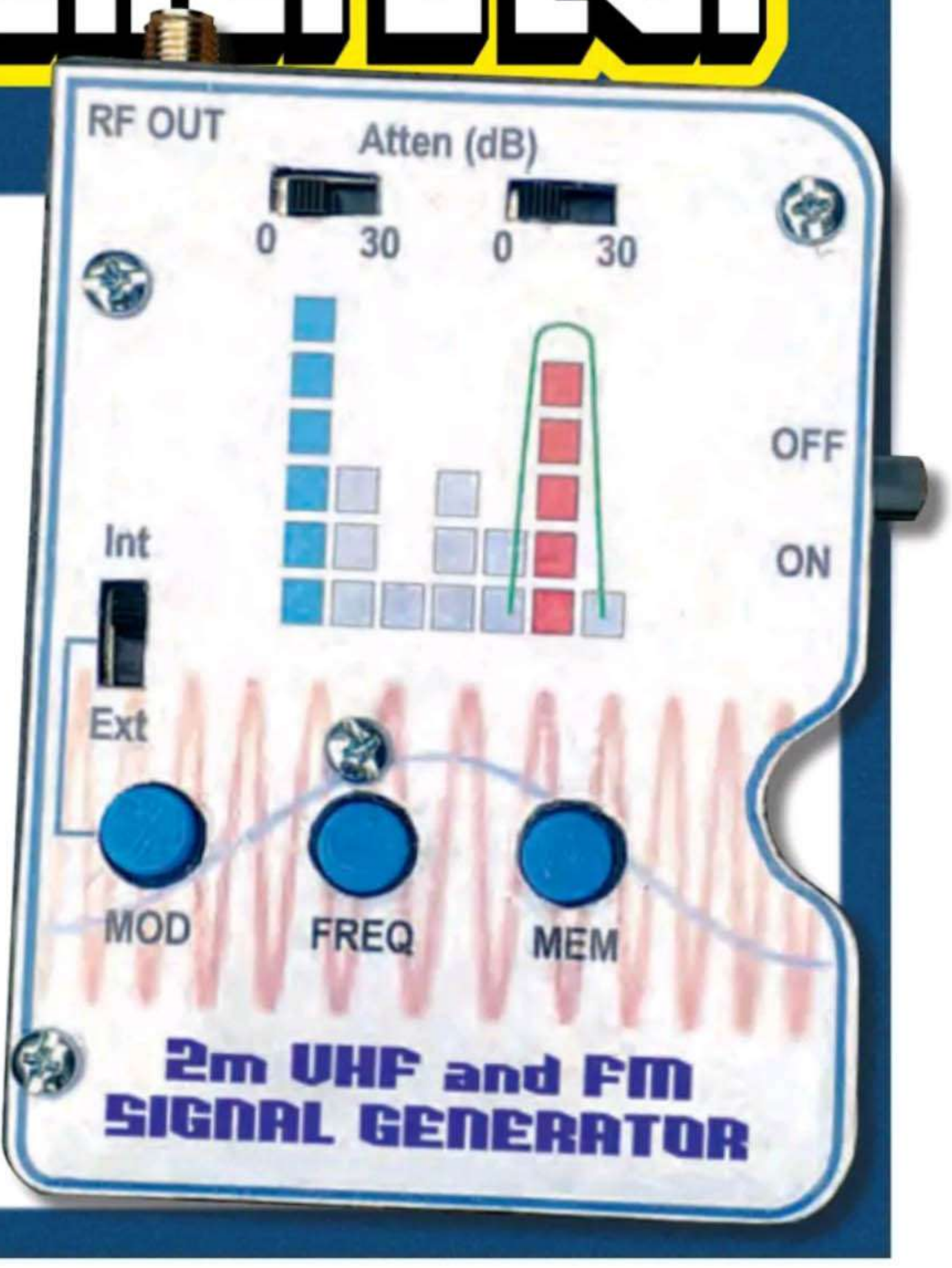
Upgrade to the new 20-year bundle for just £14.95 with: PE15to20

Download your collections at: www.electronpublishing.com

Project by Andrew Woodfield, ZL2PD

2m UHF FM Test Signal Generator

This Test Oscillator uses an AD9834 DDS chip in a somewhat unusual way, producing signals in the 2m band (144-148MHz) even though that is well above its 80MHz oscillator frequency. It can produce a CW signal at three different levels and optionally add frequency modulation.



The Analog Devices AD9834 direct digital synthesis (DDS) chip is a common feature of audio function generators and HF oscillators. This unusual variation delivers programmable FM digitally generated signals on channels across the 2m amateur radio band. It can even run from a single AA-size cell.

The 2m amateur radio band covers 144-146MHz across the UK and Europe, or 144-148MHz in North America, Australia and New Zealand.

Designs using the inexpensive AD9834 have been published for over 20 years. Yet, looking closely, almost all designs mimic the circuits from AD application notes, with the differences primarily being the choice of microcontroller, display and PCB layout.

Several years ago, I made a simple test board with an ATtiny45 microcontroller and a couple of pushbuttons. I tested the AD9834's overclocking potential and several DDS output filter designs and quickly established that the AD9834-BRUZ chips could be overclocked at up to 85MHz, a useful margin over the 50MHz specification from the data sheet.

The slightly more expensive -CRUZ devices are rated at 75MHz but are reported to handle 100MHz clocks.

That was the limit of my interest in the AD9834. I had no immediate use for it and put it on my shelf. So I expected few surprises when a friend recently asked me to confirm the results he was measuring on his AD9834 signal generator design.

Features & Specifications

- 144 to 148MHz output in 500kHz steps
- Four additional user-programmable memory frequencies in the 2m band
- –45dBm, –75dBm and –105dBm output levels
- Carrier-only (no modulation) or FM
- ±3kHz FM deviation (1kHz tone)
- CTCSS and external modulation audio inputs
- Simple to align output filter; everything else is digital

Phase modulation

As I worked on these tests, the claimed frequency modulation (FM) and phase modulation (PM) features headlined in the data sheet caught my eye. In my previous DDS designs with the AD9850 and AD9851 chips, their limited phase registers made them unsuitable for FM.

The AD9834 variable-frequency oscillator (VFO) designs I'd seen only used the DDS chip as a variable RF oscillator or audio function generator. I'd never seen any mention of AD9834 phase modulation anywhere before. Looking more closely, I realised that the AD9834 has far more capable 12-bit phase registers than the AD9850/9851. That made me curious.

Within an hour, I had phase modulation 'working' in a limited fashion. However, it wasn't clear from the data sheet how to get a specific modulation level. An extensive search of Analog Devices application information showed they were also largely silent on the topic.

As I continued to explore phase modulation, the calculations, and the other features of the chip, I realised it could achieve something quite different to the usual AD9834 design. This novel 2m CW/FM test oscillator is the result.

DDS and alias frequencies

The AD9834 is a DDS oscillator chip typical of many made by Analog Devices. The block diagram (Fig.1) shows the internal arrangement. It contains a pair of programmable Frequency Registers and a pair of Phase Registers to allow the selection of one of two combinations of frequency and phase.

These registers control a powerful numerically controlled oscillator (NCO) using the chip's 28-bit Phase Accumulator. The output of the NCO drives a lookup table and 10-bit DAC that converts the NCO output to a sinewave.

The output frequency, f_{OUT}, can be calculated as $f_{OUT} = N_{reg} \times f_{CLK} \div 2^{28}$, where f_{CLK} is the external clock frequency, and N_{reg} is the 28-bit digital value loaded into one of the two selectable frequency registers. The associated Phase Registers allow the output signal phase to be shifted by a programmable phase angle.

With an external crystal clock and a suitable low pass filter (LPF), the AD9834's output is a reasonably clean, low-noise sinewave that can reach up to about 30MHz. It's possible to generate frequencies with a resolution of 0.3Hz. The output frequency's accuracy depends on the crystal oscillator's precision and stability.

Actually, the output produced by the DDS is significantly more complex than this. In the absence of the output LPF, the DDS process also generates an extended series of signals along with the main output signal. These can reach well beyond 300MHz, as shown in Fig.2 (also from the data sheet, again with a bit of colour added).

The primary output signal (f_{OUT}) is typically filtered by an LPF that heavily attenuates these other unwanted signals. The output may be programmed to any frequency up to half the external DDS clock frequency, eg, 40MHz if the DDS uses a clock frequency of 80MHz.

Those unwanted products generated by the AD9834 are called 'alias' or 'image' signals. The first is on a frequency of $f_{CLK} - f_{OUT}$. When the DDS clock is 80MHz, if f_{OUT} is changed in steps from 1Hz to, say, 30MHz by changing the value programmed into the AD9834's Frequency Register, this first 'alias' signal (labelled $f_C - f_{OUT}$ in Fig.2) is generated.

It begins at 80MHz and reduces in frequency with each step, down to 50MHz (50MHz = 80MHz – 30MHz). These are shown in more detail in Fig.3.

If the 'wanted' f_{OUT} signal moves towards 40MHz (half the DDS clock frequency), this first alias output product becomes increasingly annoying. It also approaches 40MHz from above and must somehow be filtered out. That becomes more and more difficult as the desired f_{OUT} signal rises above 30MHz and approaches 40MHz.

This demands the use of a low pass filter with a very steep cutoff for most applications. Such filters usually start attenuating just above 30MHz, with the rejection increasing sharply to reach at least 60dB by 40MHz. Most designs use a complex 5th- or 7th-order output low-pass filter for this reason.

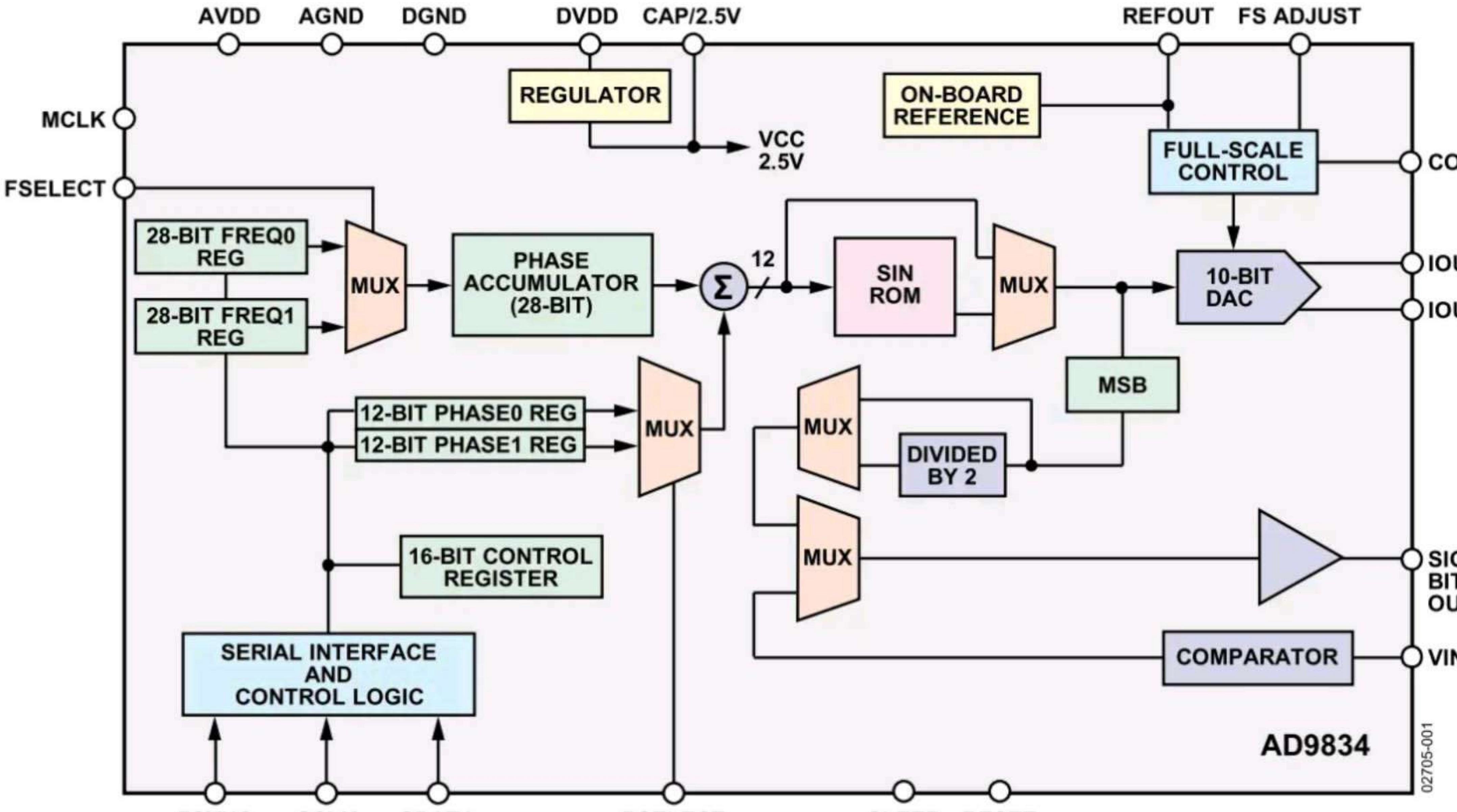

AVDD
AGND
DGND
DVDD
CAP/2.5V
REFOUT
FS ADJUST
MCLK
FSELECT
REGULATOR
VCC 2.5V
ON-BOARD REFERENCE
FULL-SCALE CONTROL
COMP
28-BIT FREQ0 REG
28-BIT FREQ1 REG
MUX
PHASE ACCUMULATOR (28-BIT)
Σ
12
SIN ROM
MUX
10-BIT DAC
IOUT
IOUTB
MSB
12-BIT PHASE0 REG
12-BIT PHASE1 REG
MUX
MUX
DIVIDED BY 2
16-BIT CONTROL REGISTER
MUX
SIGN BIT OUT
SERIAL INTERFACE AND CONTROL LOGIC
COMPARATOR
VIN
AD9834
02705-001
FSYNC
SCLK
SDATA
PSELECT
SLEEP
RESET

Fig.1: the AD9834 block diagram, reproduced from the data sheet (with some added colour). It is a typical DDS oscillator with a 28-bit phase accumulator that can generate accurate RF sinewaves up to 30MHz from an 80MHz external clock.

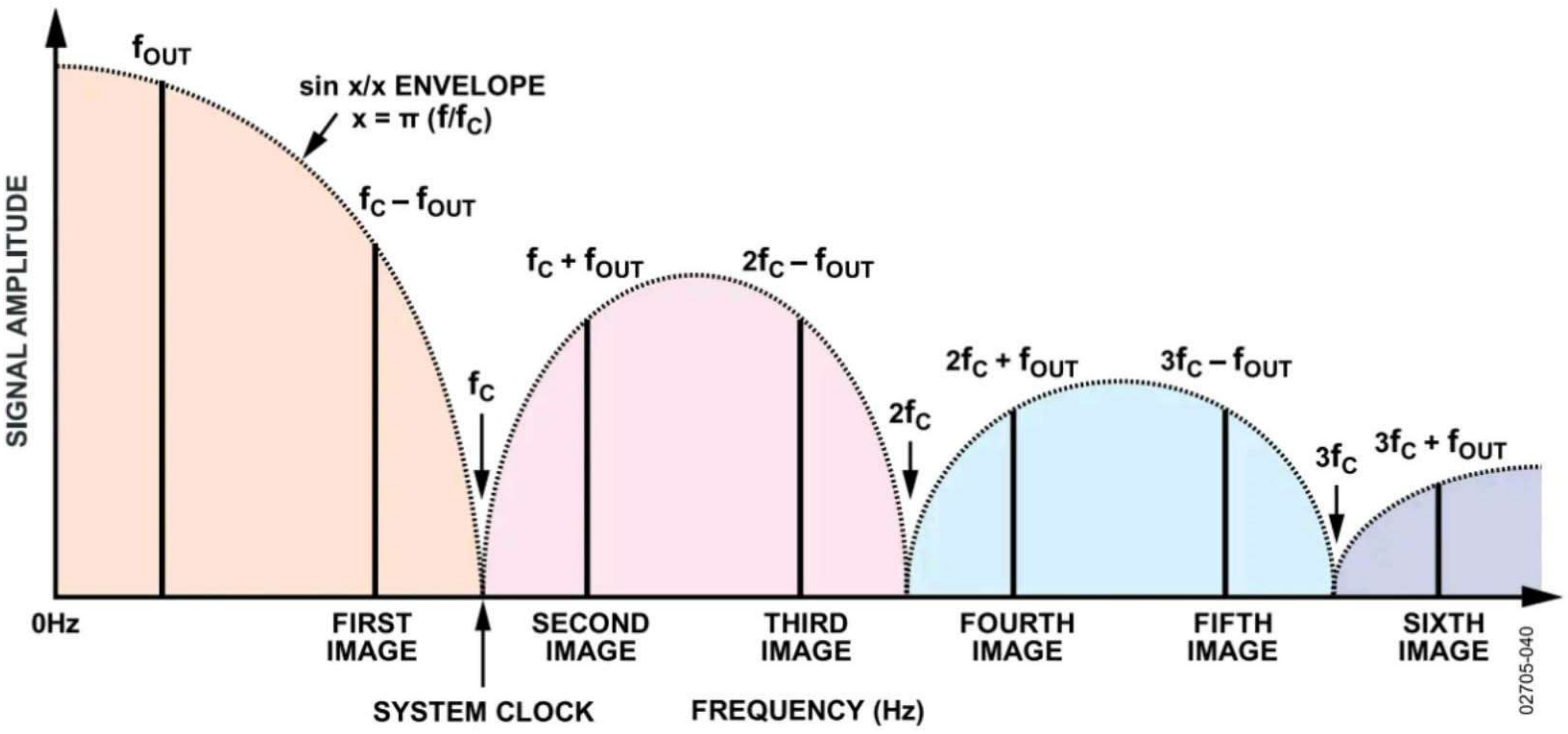


Fig.2: without any added filters, the AD9834's output signals extend well beyond 300MHz.

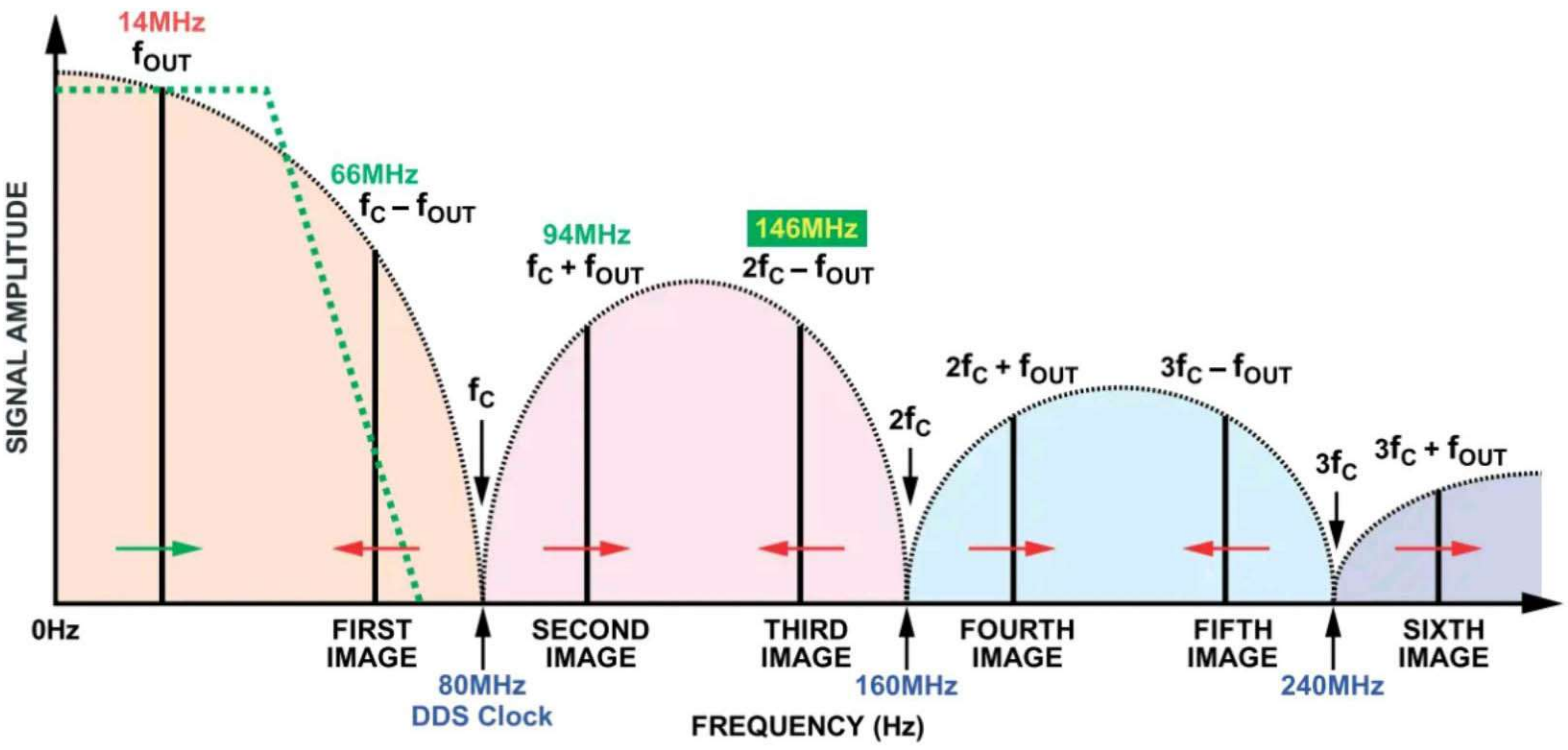


Fig.3: the AD9834 output spectrum with an 80MHz DDS clock and the frequency register set to 14MHz. The arrows show the direction the unfiltered alias carriers travel as the frequency of the 14MHz fundamental increases. The green dashed line shows the typical high-order low-pass filter normally used to remove these other products from the output signal.

You can also see from Fig.3 that the DDS output level is not the same across the spectrum. The output level falls following a sin(x) ÷ x response. With an 80MHz clock, for example, the output at 30MHz is 2dB less than at 1MHz.

The 50MHz 'alias' output generated when the main output is 30MHz is 6.5dB below the 1MHz level, and only 4.5dB below the 30MHz wanted output. So, without a good LPF, the AD9834 output at 30MHz would have severe distortion due to the desired signal mixing with the nearby unwanted 50MHz alias output product.

Operating the AD9834 in the 2m band (144-148MHz)

Fig.3 shows the other alias signals generated above the desired and first alias outputs. If the desired output is 14MHz and the first alias is 66MHz, the next alias output is 94MHz. More alias outputs are generated at 146MHz, 174MHz, 226MHz and beyond.

The AD9834 output also contains a residual clock output at 80MHz. Since it is a square wave clock, it has a strong unwanted product at the third clock harmonic, 240MHz in this case. Fig.3 also shows the direction the aliases move as the main carrier increases in frequency.

Alias outputs can appear in the 2m band, for example, if the AD9834 output is set in turn to 16MHz, 15MHz, 14MHz, 13MHz and 12MHz with an 80MHz DDS clock, the third-image alias output ('super-Nyquist' product) will be generated at 144MHz, 145MHz, 146MHz, 147MHz and 148MHz respectively, albeit at much lower levels.

If a relatively narrow bandwidth LC bandpass filter is added to the AD9834 output and tuned to the 2m band, passing just these 2m-band signals is possible. An example of the response of such a filter for use with the AD9834 is shown in Fig.4.

This output filter must also be designed to match the 200Ω output impedance of the AD9834 and allow for a 50Ω output load impedance, to match the expected loads in typical RF applications.

Modulation

With this filter selecting the 2m band signals from the AD9834 output, the next step was to see if it was possible to achieve frequency modulation with the chip. As noted earlier, the AD9834 data sheet highlights the possibility of phase (and thus frequency) modulation but gives no further detail.

Analog Devices' application notes did not provide any further details about how the AD9834 phase modulation registers might be used to achieve this. Also, despite a thorough search, I could not find any DDS design in which this feature was actually used.

That led me to dig further into phase modulation. I analysed and tested the AD9834 Phase Registers to understand their impact on the DDS output signal.

When FM was emerging on the 2m VHF amateur radio band from 1970 to 1980, phase modulation (PM) was generally considered the preferred approach. There were claims of "better quality" modulation and "more natural sounding" voices. However, there was little to suggest any evidence supporting these claims.

Practical issues, and the arrival of cheap varicap diodes, led to PM being quickly overwhelmed by FM. Varicaps were easy to use in oscillators and often reduced the component count, unfortunately sometimes at the expense of modulation linearity.

PM quickly fell out of favour, and that may have led to the minimal information about phase modulation in the technical magazines, handbooks and reference textbooks of the period.

One useful source from those days was William Orr's classic "Radio Handbook". The 1981 edition briefly described the method and provided a few examples; see my summary in Fig.5.

First, the transmitter's frequency deviation (ie, modulation) is directly proportional to the amplitude of the input audio signal level for both phase modulation and frequency modulation.

With FM, the frequency deviation remains constant regardless of the input signal's frequency. However, with PM, the deviation increases with increasing frequency.

Since phase-modulated transmitters were initially more popular, the characteristic PM frequency response, later referred to as 'pre-emphasis', also required the reverse audio frequency characteristic to be implemented in

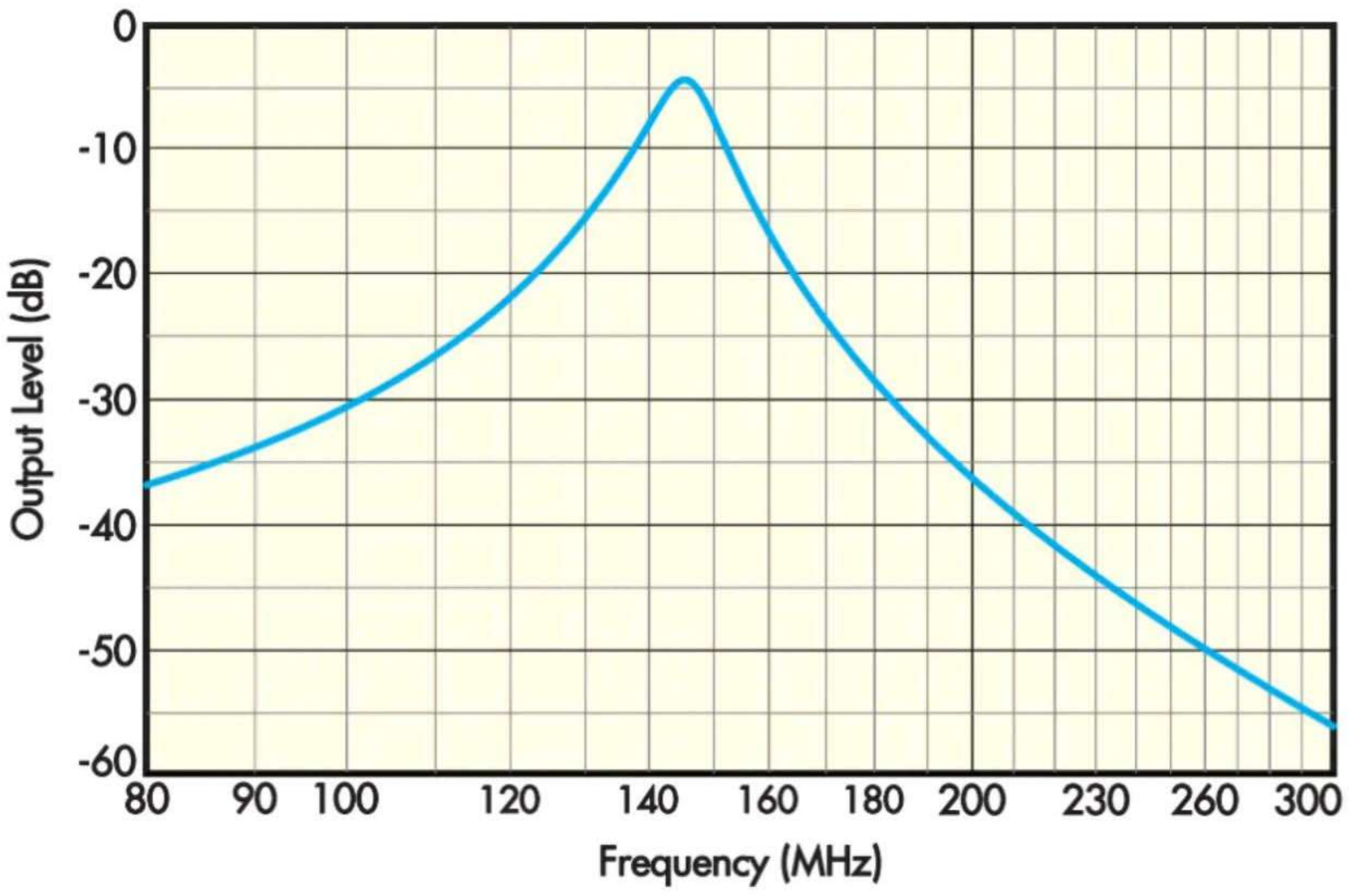


Fig.4: a bandpass filter using discrete inductors and capacitors can give this response, which allows the 144-148MHz alias output product to be selected while other alias, clock and fundamental signals are rejected.

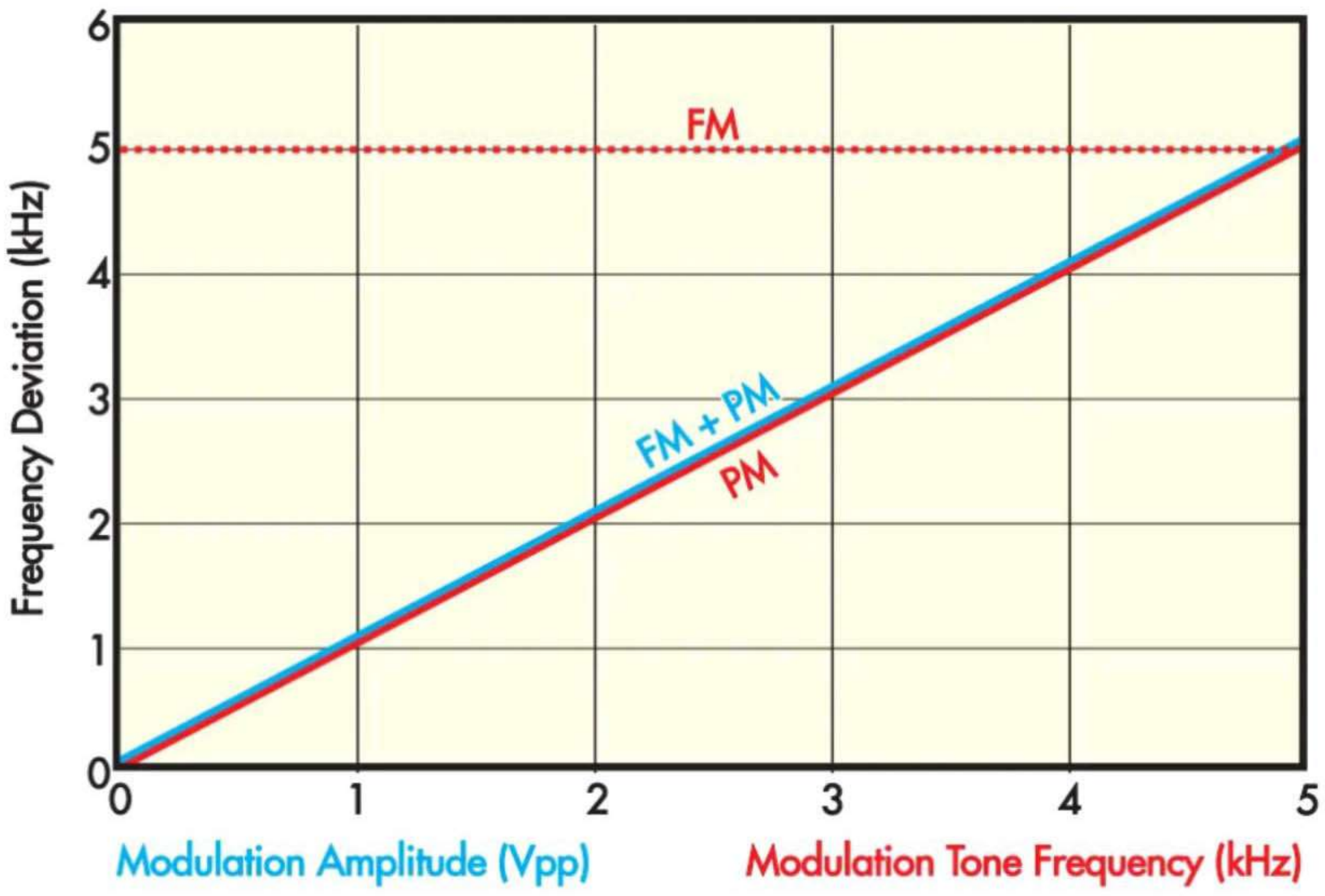


Fig.5: the frequency deviation of a phase- or frequency-modulated transmitter depends on the modulation level, but phase-modulated transmitter deviation also depends on the input modulation frequency. (Adapted from Bill Orr's "Radio Handbook", 1981)

the receiver, ie, 'de-emphasis'. This is usually achieved by a simple RC circuit located immediately after the FM receiver's discriminator (detector) stage.

FM transmitters required the addition of this pre-emphasis characteristic to work correctly with those FM receivers. A similar RC circuit was usually added just ahead of the FM modulator to mimic phase modulation. Since the effect of noise increases with audio frequency, adding pre-emphasis to FM (or just using phase modulation with its integral pre-emphasis characteristic) improves noise performance.

AD9834 phase modulation

With that background, let's return to the AD9834. Phase modulation in the AD9834 is produced by making periodic changes to the value stored in the Phase Register (PHASE0/1 REG; see Fig.1). The Phase Register's value results in a precise phase shift of the current DDS output signal.

The DDS output frequency is determined by the value in the AD9834's Frequency Register. The AD9834's Phase Register value shifts the phase of the f_{OUT} carrier by 2π ÷ 4096 multiplied by the value contained in the Phase Register. The Analog Devices data sheet doesn't explicitly state this, but that's what it does.

As to the lack of any application of this information, any example, or further supporting detail, AD9834 users are left to fathom the usefulness of this relationship for themselves. As it turns out, by periodically storing a value proportional to the amplitude of an incoming audio signal in the AD9834's 12-bit Phase Register, it is possible to produce the desired PM (and thus FM) signal.

There should, by rights, be a fanfare of trumpets at this point in the story, but there's another crucial detail. The phase shift that produces phase modulation in the AD9834 is the same at the fundamental output frequency as for all the other aliased carriers (see Fig.3).

That is entirely different from the traditional phase modulators and transmitters described in reference books, those early FM broadcast transmitters, and the really old, sorry, 'legacy' 2m VHF ham transmitters.

These traditional transmitters used a series of frequency multiplier stages to generate the required VHF carrier (before the arrival of phase-locked loops [PLLs]). A typical early 2m VHF ham transmitter might have a 12MHz crystal oscillator followed by a series of multiplier stages.

AD9834 phase modulation details

As phase deviation is proportional to both the frequency and amplitude of the modulating signal, the following equation can be used:

Frequency deviation (in kHz) = phase shift (in radians) × modulation frequency (in kHz)

For example, if the modulating signal's frequency is 1kHz and we have a carrier phase shift of +½ radian, the resulting output signal's frequency deviation is +500Hz (note that 2π radians = 360°).

A standard signal generator setup for testing a 2m amateur radio VHF FM receiver (25kHz channel spacing) uses a 1kHz test tone and a carrier frequency deviation of ±3kHz. Therefore, we require a maximum phase shift on the AD9834 output carrier of 3kHz ÷ 1kHz = 3 radians.

The 12-bit Phase Register in the AD9834 generates a phase shift (on the DDS output carrier) of π radians when PHASEREG = 2048. Therefore, to achieve 3 radians of phase deviation, the Phase Register must be loaded with a peak value of 3 ÷ π × 2048 = 1956.

The 1kHz internal oscillator delivers a 3.7V peak-to-peak sinewave to the micro's ADC0 analog input. This gives a peak ADC value in the 10-bit ADC register in the ATtiny45 of about 750. The ADC reference voltage is 5V, so 5V at the ADC input would result in a maximum reading of 1023.

The software scales this 750 input value to give a peak Phase Register value of about 2250. That is a little higher than the calculated value of 1956 due to rounding errors in the simple integer calculation routine used. Tests with a professional-grade modulation meter confirmed this value produced ±3kHz deviation in the AD9834 output signal.

If you use the external modulation input in this design, the maximum frequency deviation that can be achieved is about ±4.5kHz. This is due to the ADC measurement limit of 1023 (because of the 10-bit ADC in the ATtiny45) with a 5V peak-to-peak audio input. That input level must not be exceeded, or the ATtiny45 could be damaged.

They usually used a frequency tripler stage followed by two frequency doublers to multiply the oscillator frequency by 12, giving a final output at 144MHz. Consequently, the frequency deviation measured at 144MHz was twelve times that at 12MHz.

That was very useful because these legacy phase and frequency modulators could not be phase-shifted (or frequency-shifted) very much. But, since the transmitter's multiplier stages also multiplied the phase and frequency modulation deviation, the final output readily achieved the desired modulation deviation.

It's a different situation with a DDS. If AD9834 output (f_{OUT}) is set to 14MHz, changing the Phase Register values appropriately in the AD9834 will produce a 1kHz tone with a 3kHz frequency deviation on that output. The 146MHz alias product will have an identical 3kHz frequency deviation due to the aliasing process in the DDS.

There is no 'multiplication effect' like that in those traditional PM and FM transmitters. The alias outputs are all directly generated equally and simultaneously by the DDS NCO process.

Furthermore, the AD9834's phase modulator is a 100% digital process carried out in the NCO. The maximum frequency deviation is limited by the Phase Register length and the NCO process.

The accompanying panel titled "AD9834 phase modulation details" describes how the Phase Register values generate the desired PM (and FM) deviation.

Circuit details

Fig.6 shows the circuit of this compact 2m FM Test Generator.

The AD9834 is controlled by an Atmel/Microchip 8-pin ATtiny45 microcontroller using a three-wire SPI serial bus. The ATtiny45 has a hardware SPI interface, simplifying the software and increasing data transfer speed. This SPI interface only uses two (USCK and DO) of the usual three SPI lines because no data needs to be read from the AD9834.

The ATtiny45 is clocked at 16MHz using its internal RC oscillator and PLL. That releases all six I/O pins for this design (the other two are the 5V power supply).

Practically all of the firmware operates using a series of software interrupt routines. Three pushbuttons (S1-S3) control the generator's frequency and modulation. These all connect to pin 2 of the ATtiny45. Pressing any button triggers a software interrupt routine that measures the voltage at that pin to determine which button was pressed.

The Frequency button (S2) allows the selection of one of eight fixed channels at 500kHz intervals from 144 to 148MHz. The Memory button (S3) selects one of four user-programmable channels in the 144-148MHz range. Finally, the Modulation button (S1) turns the 1kHz modulation tone on and off. As described in the panel, this produces a frequency deviation of ±3kHz.

A modified interrupt routine could support a rotary encoder for multi-channel frequency tuning, memory selection and other features. However, without a suitable display – there are just not enough pins – the design was intentionally kept ultra-simple and inexpensive.

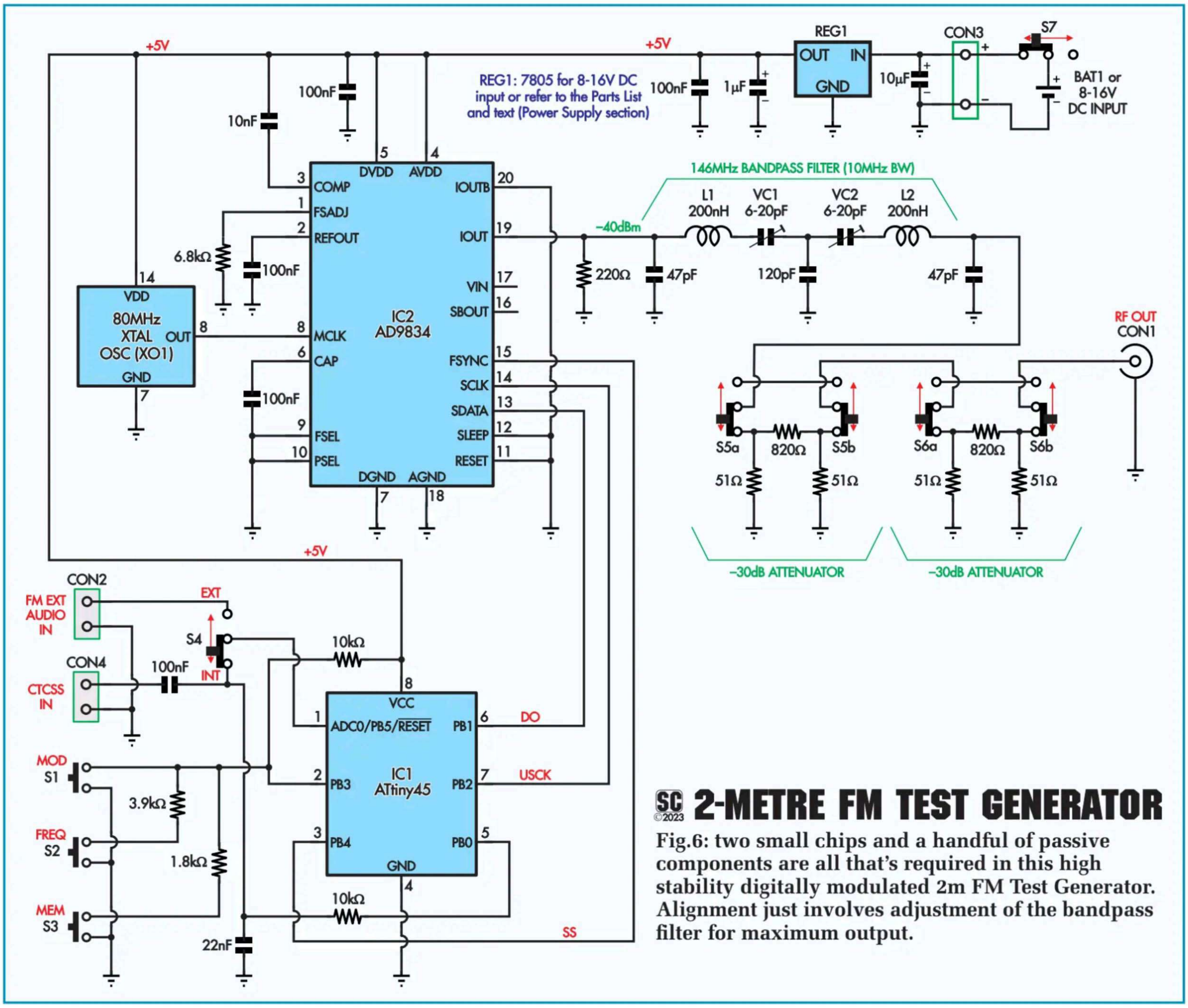
Generating a test tone

The internally-generated 1kHz modulation test tone is produced by 'bit-banging' digital output pin 5 (PB0). Usually, one of the ATtiny's internal timers would be used to do this. However, that wasn't possible here because the timer-related pins were already handling the AD9834 SPI control bus.

The bit-banging process produces an unusual low-harmonic PWM output. It is designed to null the 5th and 7th harmonics. As a result, it only requires a very modest 10kΩ/22nF RC filter to give a remarkably clean and accurate 1kHz sine wave. The sinewave measures around 3.7V peak-to-peak (with V_{CC} at 5V) when it arrives at the ATtiny45's ADC0 input on pin 1.

The 1kHz PWM tone is generated continuously at pin 5. However, it is only sampled (after the RC filter) at pin 1 with the ATtiny's analog-to-digital converter (ADC) when modulation is required. The sinewave's amplitude is sampled 8000 times each second (8ksps). These samples are used to update the Phase Register in the AD9834.

The relationship between the audio tone's amplitude and frequency and the DDS-generated frequency deviation is explained in the panel above.


+5V
+5V
REG1
OUT IN
GND
CON3
S7
+
−
10µF
BAT1 or
8-16V
DC INPUT
REG1: 7805 for 8-16V DC input or refer to the Parts List and text (Power Supply section)
100nF
1µF
100nF
10nF
146MHz BANDPASS FILTER (10MHz BW)
L1 200nH
VC1 6-20pF
VC2 6-20pF
L2 200nH
−40dBm
220Ω
47pF
120pF
47pF
RF OUT
CON1
DVDD
AVDD
COMP
FSADJ
REFOUT
IOUTB
IOUT
VIN
SBOUT
IC2
AD9834
MCLK
CAP
FSYNC
SCLK
SDATA
SLEEP
RESET
FSEL
PSEL
DGND
AGND
6.8kΩ
100nF
100nF
VDD
80MHz
XTAL
OSC (XO1)
OUT
GND
S5a
820Ω
S5b
S6a
820Ω
S6b
51Ω
51Ω
51Ω
51Ω
−30dB ATTENUATOR
−30dB ATTENUATOR
+5V
CON2
FM EXT AUDIO IN
EXT
S4
INT
CON4
CTCSS IN
100nF
10kΩ
VCC
ADC0/PB5/RESET
PB1
DO
PB3
IC1
ATtiny45
PB2
USCK
PB4
PB0
GND
MOD S1
FREQ S2
MEM S3
3.9kΩ
1.8kΩ
22nF
10kΩ
SS

SC ©2023 2-METRE FM TEST GENERATOR

Fig.6: two small chips and a handful of passive components are all that's required in this high stability digitally modulated 2m FM Test Generator. Alignment just involves adjustment of the bandpass filter for maximum output.

2m VHF FM Test Signal Generator

The ±3kHz deviation for FM gives the usual 60%-of-peak-modulation level used for testing 25kHz VHF FM channels.

By making this modulation process external to the ATtiny45, external modulation sources can also be used. If external modulation is selected with S4, the maximum input level of 5V peak-to-peak will produce at most ±4.5kHz deviation.

For those using 12.5kHz FM channels, an external audio level of 3V peak-to-peak will give ±2.5kHz deviation with the Test Generator (peak deviation), and 2V peak-to-peak will give ±1.75kHz deviation (60% of peak deviation). Those figures all assume V_{CC} is close to 5V.

The 8kHz sampling rate used in this design limits the modulating frequency to less than 4kHz (the Nyquist limit). However, the lack of any anti-aliasing filter in the software practically limits external modulation frequencies to less than 3kHz.

Output signal generation

An 80MHz external crystal oscillator clocks the AD9834. This frequency is close to optimal for this application because it eases the RF filtering task slightly, and suitable crystal oscillators are readily available at low cost.

As described earlier, a low-loss, highly selective bandpass filter is required to extract the wanted super-Nyquist 2m-band RF signal. A pair of high-Q air-wound inductors are used. These are essential to produce the desired result. The filter is designed to give about 40dB of attenuation at the nearest alias bands close to 100MHz and 200MHz while introducing no more than 4dB passband loss.

Happily, these inductors are quick and easy to make at minimal cost.

Despite the wide variation in output levels generated by the DDS process, the resulting 2m band output levels (post-filter) are about -40dBm ±2dBm. The directly generated carrier, the residual clock and the other aliases are attenuated by 25dB or more, and spurious products are at least 30dB below the output level.

A pair of switched attenuators provide three output levels suitable for receiver tests. The highest output level, -45dBm ±2dBm, places an FM receiver well into limiting without overloading, producing a 1kHz demodulated audio tone with a very good signal-to-noise ratio.

Enabling one attenuator (either) gives a signal level of about -75dBm. This is close to the typical ‘corner’ of FM receiver performance where limiting begins to improve the receiver’s signal-to-noise usefully. Adding the second attenuator gives a test signal of about -105dBm. This is close to that used in typical 12dB SINAD receiver sensitivity and squelch gating tests.

The absolute accuracy of these levels depends in part on the output filter alignment and the construction method. The prototype was housed in a 3D-printed enclosure, which provides limited shielding. That limits the absolute accuracy of the signal level and the absolute accuracy of some measurements, so if you’re after precision, you will need a metal case for shielding.

Power supply

My initial plan was to derive the 5V supply for the ATtiny45 and AD9834 using a 7805 linear regulator.

While the AD9834 data sheet states it is a “low current DDS device” (20mA at 5V), it needs an external clock generator, in this case at 80MHz. These typically consume 30-70mA, although a few will operate with as little as 10mA. Hence, a 78L05 might not be sufficient when operating from a 9-12V DC supply.

An alternative is to use a switch-mode regulator module like the one shown in Photo 1. This can be mounted in place of the 7805 linear regulator on the PCB. The input voltage can be from 6V to 15V DC, and the 5V output can deliver up to 500mA. It improves efficiency and remains cool during operation.

The prototype is powered by a single 1.5V AA alkaline cell. This is only suitable for intermittent use due to the limited capacity of the cell but allows a compact 3D-printed PLA enclosure to be used. A boost regulator module steps the cell voltage up to 5V (shown in Photo 2).

Finally, powering the Generator from a single Li-ion or LiPo cell is also possible. They have a nominal fully charged output voltage of 4.2V and an operating end-point voltage of 3.5V. I tested the Generator with supply voltages from 3-5V. The output level remained constant within 0.2dB across that voltage range!

If using a Li-ion or LiPo cell, there is also the option to integrate a small USB charger PCB (see Photo 3). Some versions include an automatic battery disconnect feature to ensure the battery does not operate below 3.5V, which could damage it.

Construction

The Test Generator is built on a small 50 × 70mm PCB, coded 06107231, that hosts a mix of SMD and through-hole parts. The board has a near-continuous top-side ground plane with all the SMD parts mounted on the underside. This arrangement keeps the unit compact while allowing for easier testing and modification during development. It also produced improved RF performance over other approaches.

The layout is shown in Fig.7. Start by fitting the 23 SMD components on the underside. It is easiest to start with the AD9834. Position it over its pads, tack one lead, then double-check that its pin 1 orientation is correct and all pins are correctly centred on their pads before soldering the rest. Adding a little bit of flux paste to the pads and on the leads will make soldering it much easier.

If you accidentally bridge any of its pins (which is easy to do), add

Photo 1: a DD4012SA small switching buck regulator (top and bottom shown) can deliver 5V DC to run the chips efficiently.

Photo 2: this tiny 5V boost regulator module is used to step up the cell voltage.

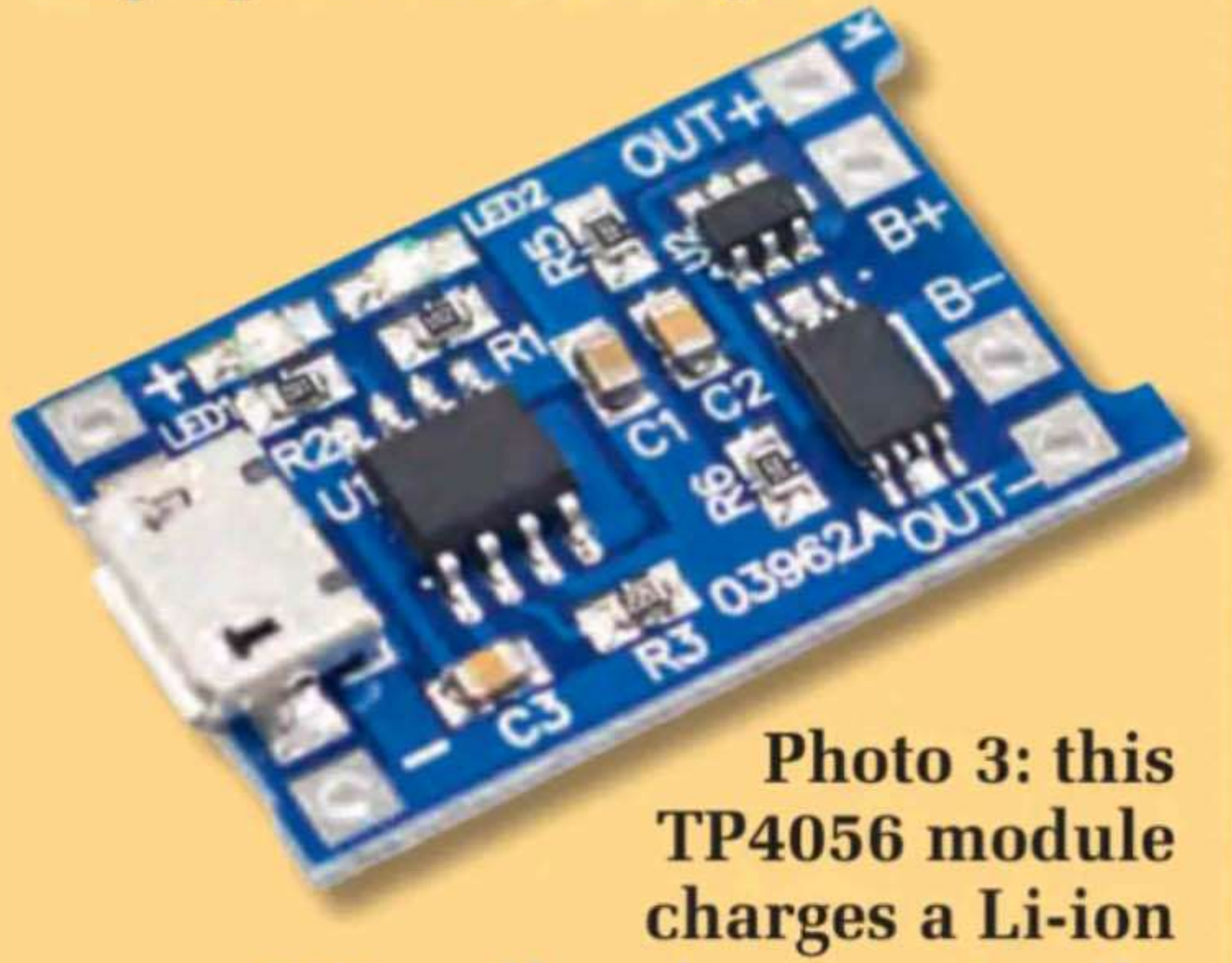


Photo 3: this TP4056 module charges a Li-ion or LiPo cell and automatically disconnects the load if the terminal voltage falls too low.

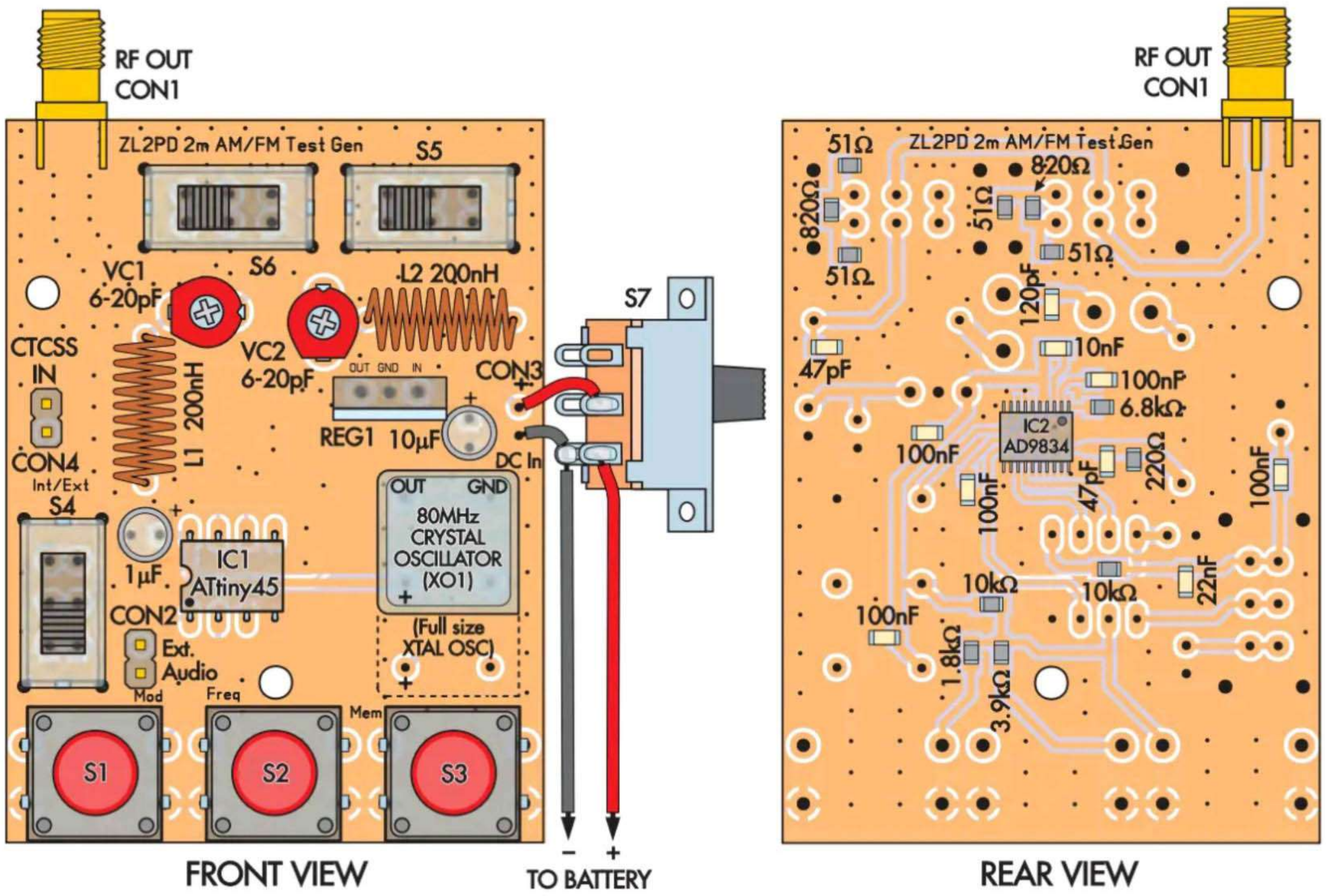


Fig.7: the Test Generator is built on a compact 50 × 70mm double-sided PCB with SMDs on the underside and the through-hole components on the top.

a bit of flux paste to the bridge and apply some clean solder wick with your soldering iron. Once it gets hot enough and the flux starts to smoke, the excess solder will be pulled into the wick, leaving clean solder joints without a bridge. Repeat as necessary until all the solder joints look good under magnification.

The remaining SMDs can then be fitted. All can be soldered in place by hand with a fine-tipped soldering iron. Fit the edge-mount SMA coaxial connector after that.

Make the two inductors using 0.4mm diameter (26SWG) enamelled copper wire wound on a 5mm diameter drill bit shaft. Close-wind 10 turns for each, then stretch each coil slightly until each measures 13mm long. Allow 10mm of extra wire at each end of each coil for the connections.

While keeping each coil on the drill bit, scrape the enamel from the wire ends to allow for the soldered PCB connections.

Flip the board over and continue construction by adding the ATtiny45's socket (watching its orientation), the two trimmer capacitors, the two inductors and the two electrolytic capacitors on the top side of the PCB.

Now mount the crystal oscillator module. The PCB allows for either full or half-sized oscillator modules to be used. Next, install the three pushbuttons, the two slide switches, the toggle switch, and finally, the regulator (a 7805 or one of the other options).

Generating the EEPROM file

I have created a spreadsheet to allow the easy entry and programming of the four user-selected 2m frequencies. These may be on any frequency from 144MHz to 148MHz. Enter your choice of frequencies in the blue cells. It's best not to touch anything else!

After entering the four memory channel frequencies into the spreadsheet provided (left side of Screen 1). Scroll down to the bottom of the worksheet and click on the green Write EEP File button (Screen 2). The spreadsheet then generates and saves the 2mTestGenFreq.EEP file in the same directory as the spreadsheet file.

If you don't have Excel, you can open the file in a free package like LibreOffice, and everything will work except for the final file-saving step; pressing the green button will do nothing. Instead, after updating the frequencies, check the text just to the right of that button. You will see three lines that start with colons.

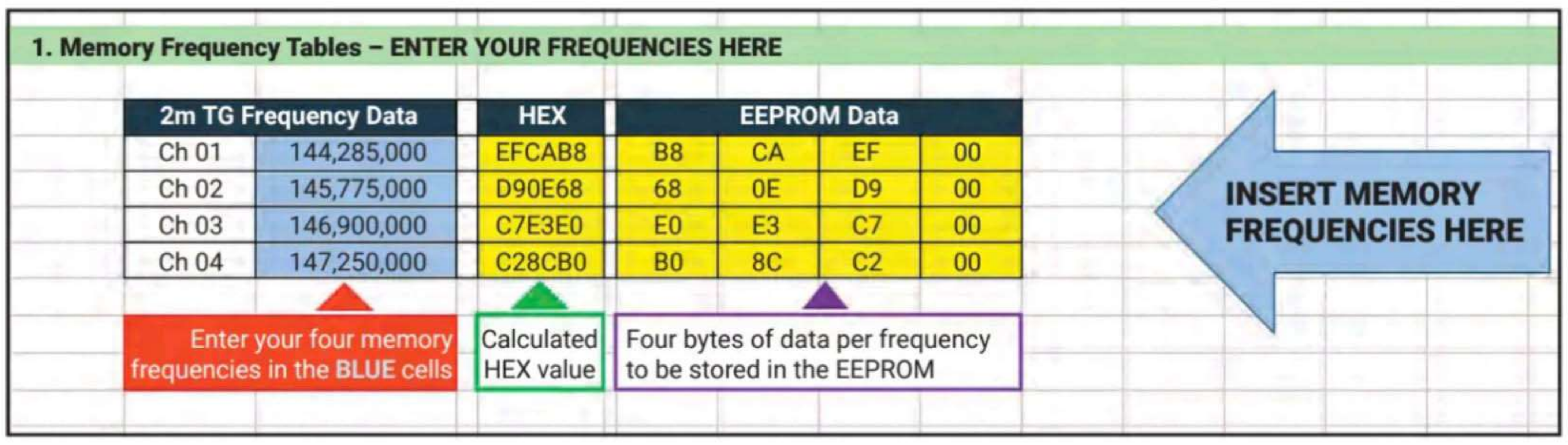
Click your mouse on the left side of the first line that starts with a colon (just to the right of the colon), then drag it down to the third line and release the button. Press CTRL+C (or the equivalent command to copy to the clipboard), then create a new text file, open it and press CTRL+V (to paste those lines into it). Save that file and then rename it from a .txt extension to .eep.

That gives the same result as Excel does when pressing the button.

Either way, rename the resulting EEP file so you know what it's for. Otherwise, the next time you use the spreadsheet, it will overwrite your previous file.

Programming the ATtiny45

The HEX file for the Test Generator is available for download from the *Practical Electronics* website, along with



1. Memory Frequency Tables – ENTER YOUR FREQUENCIES HERE

2m TG Frequency Data		HEX	EEPROM Data			
Ch 01	144,285,000	EFCAB8	B8	CA	EF	00
Ch 02	145,775,000	D90E68	68	0E	D9	00
Ch 03	146,900,000	C7E3E0	E0	E3	C7	00
Ch 04	147,250,000	C28CB0	B0	8C	C2	00

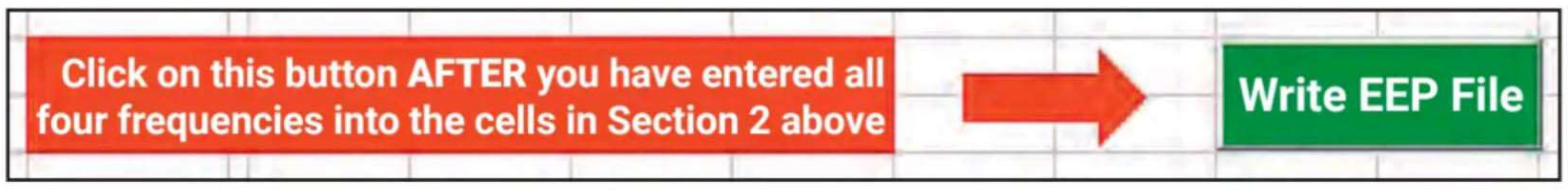
Enter your four memory frequencies in the BLUE cells

Calculated HEX value

Four bytes of data per frequency to be stored in the EEPROM

INSERT MEMORY FREQUENCIES HERE

Screen 1: a shot of the spreadsheet which is used to generate the data required to program the user-selected 2m frequencies.



Click on this button AFTER you have entered all four frequencies into the cells in Section 2 above

Screen 2: clicking on the green button automatically generates the ATtiny45 EEP file, which contains your four desired frequencies.

Fuse settings for the ATtiny45

Byte	Value	Notes
Lock byte	0xFF	
Extended byte	0xFF	
High byte	0x57	RSTDISBL = 0 (set), EESAVE = 0 (set)
Low byte	0xE1	CKSEL = 0001 for 16MHz internal RC oscillator, CKDIV8 = 1 (disabled)

the BASCOM source code. You can load the EEP file with the four user-defined frequencies into the chip at the same time.

Program your ATtiny45 with the HEX and EEP files using a suitable programmer (USBasp etc). After that, program the configuration fuses. Table 1 ("Fuse settings for the ATtiny45") shows the required fuse settings.

My article on the Shirt Pocket DDS Oscillator in the September 2021 issue included a small programming adaptor that can be used to program an ATtiny chip out of the circuit in conjunction with a suitable serial programmer.

You could also use our May & June 2013 PIC/AVR Programming Adaptor, or build an adaptor on a breadboard or small piece of protoboard.

Reprogramming the ATtiny45

If you want to change the memory channel frequencies, you can't just put the chip back into a regular programming adaptor since the $\overline{\text{RESET}}$ pin is disabled. The ATtiny45 must first be erased using a special HV programmer.

I have designed a simple chip eraser and fuse restorer ("CEFR") to do this. You can read about how to build it on my website at **www.zl2pd.com/CEFR_Fuse_Reset_Tool.html** – it requires no special parts and can be powered from a USB socket or external 5V USB power supply. Another well-known DIY fuse resetting tool is the Fuse Doctor (see **https://github.com/SukkoPera/avr-fusebit-doctor**).

Final assembly

Depending on the enclosure and power supply option you select, add the power switch and power wiring to suit. In the prototype, the AA cell is located in the lower part of a 3D-printed

Parts List – VHF / FM Test Signal Generator

1 double-sided PCB coded 06107231, 50 × 70mm
1 3D-printed case & front panel label
1 set of AA cell contacts
1 AA alkaline cell
1 80MHz crystal oscillator, full or half-size DIP type (X01)
  [AliExpress pemag.au/link/abmb]
3 PCB-mount 4.3mm momentary tactile pushbutton switches (S1-S3)
4 solder tag miniature DPDT slide switches (S4-S7)
  [Altronics S2010, Jaycar SS0852]
1 8-pin DIL IC socket (for IC1)
1 SMA edge connector (CON1)
3 2-pin headers, 2.54mm pitch (**optional**) (CON2-CON4)
1 3-pin right-angle header, 2.54mm pitch (if REG1 has none)
2 No.4 × 5mm self-tapping screws
2 M3 x 8mm panhead machine screws
2 M3 x 8mm countersunk head machine screws
2 10mm-long M3-tapped Nylon spacers
1 400mm length of 0.4mm diameter/26SWG enamelled copper wire
  (for L1 & L2)
various lengths of light-duty hookup wire

Semiconductors
1 ATtiny45-20PU 8-bit microcontroller programmed with
  0610723A.HEX, DIP-8 (IC1)
1 AD9834-BRUZ or -CRUZ DDS signal generator IC, TSSOP-20 (IC2)
  [AliExpress pemag.au/link/abmc]
1 3-pin 5V output boost module (REG1) (for AA cell operation)
  [AliExpress pemag.au/link/abmd] ***OR***
1 3-pin 5V output buck module (REG1) (for 6.5-40V DC operation)
  [AliExpress pemag.au/link/abme] ***OR***
1 7805 5V 1A linear regulator, TO-220 (for 8-16V DC operation)

Capacitors *(all SMD M2012/0805 50V ceramic unless noted)*
1 10μF 50V/63V radial electrolytic
1 1μF 50V/63V radial electrolytic
5 100nF X7R
1 22nF X7R
1 10nF X7R
1 120pF NP0/C0G
2 47pF NP0/C0G
2 6-20pF PCB-mount trimmer capacitors (VC1, VC2)

Resistors *(all 1% SMD M2012/0805 size)*
2 10kΩ
4 51Ω
1 6.8kΩ
1 220Ω
1 3.9kΩ
1 1.8kΩ
2 820Ω

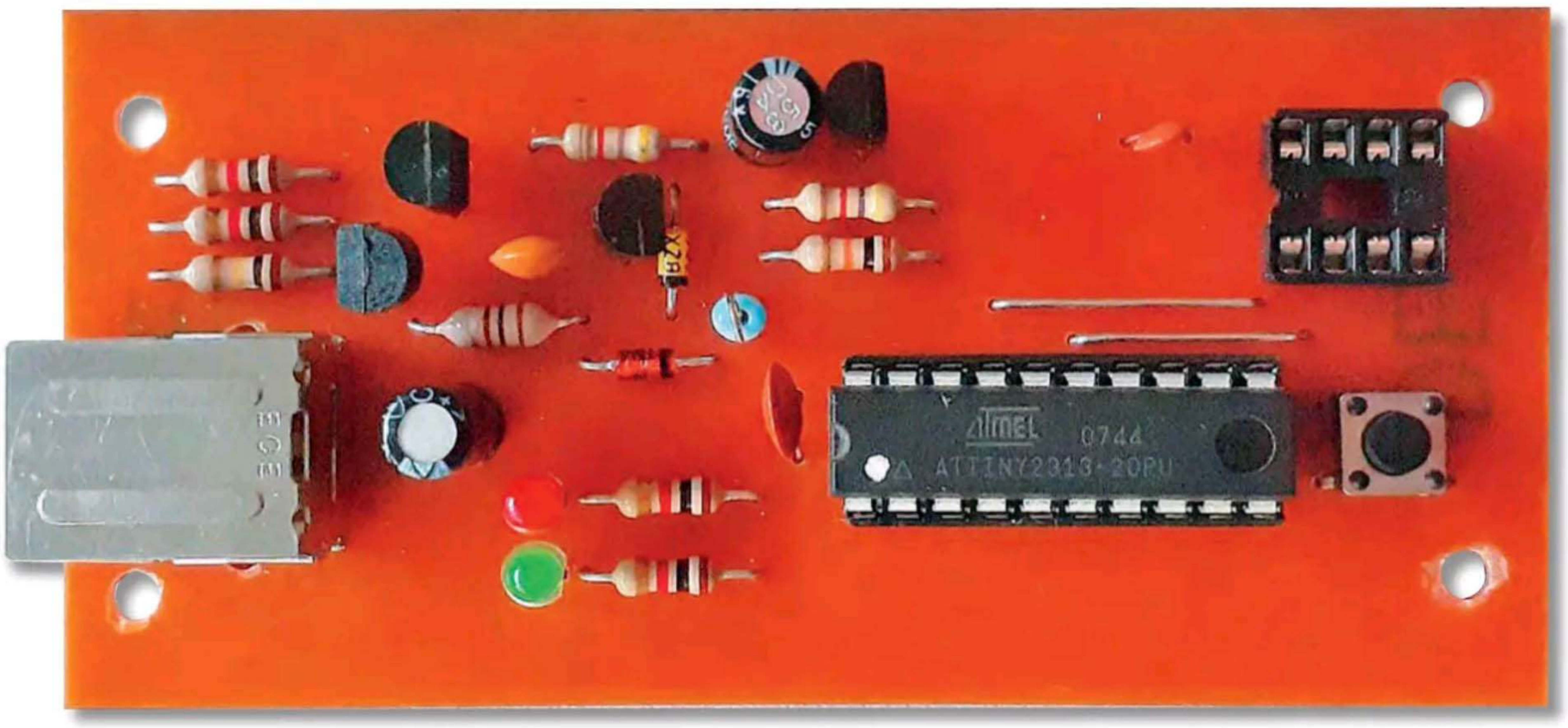


Photo 4: my simple Chip Eraser and Fuse Restorer (CEFR) resets ATtiny25/45/85 fuses back to the factory default settings so that you can change the memory channel frequencies.

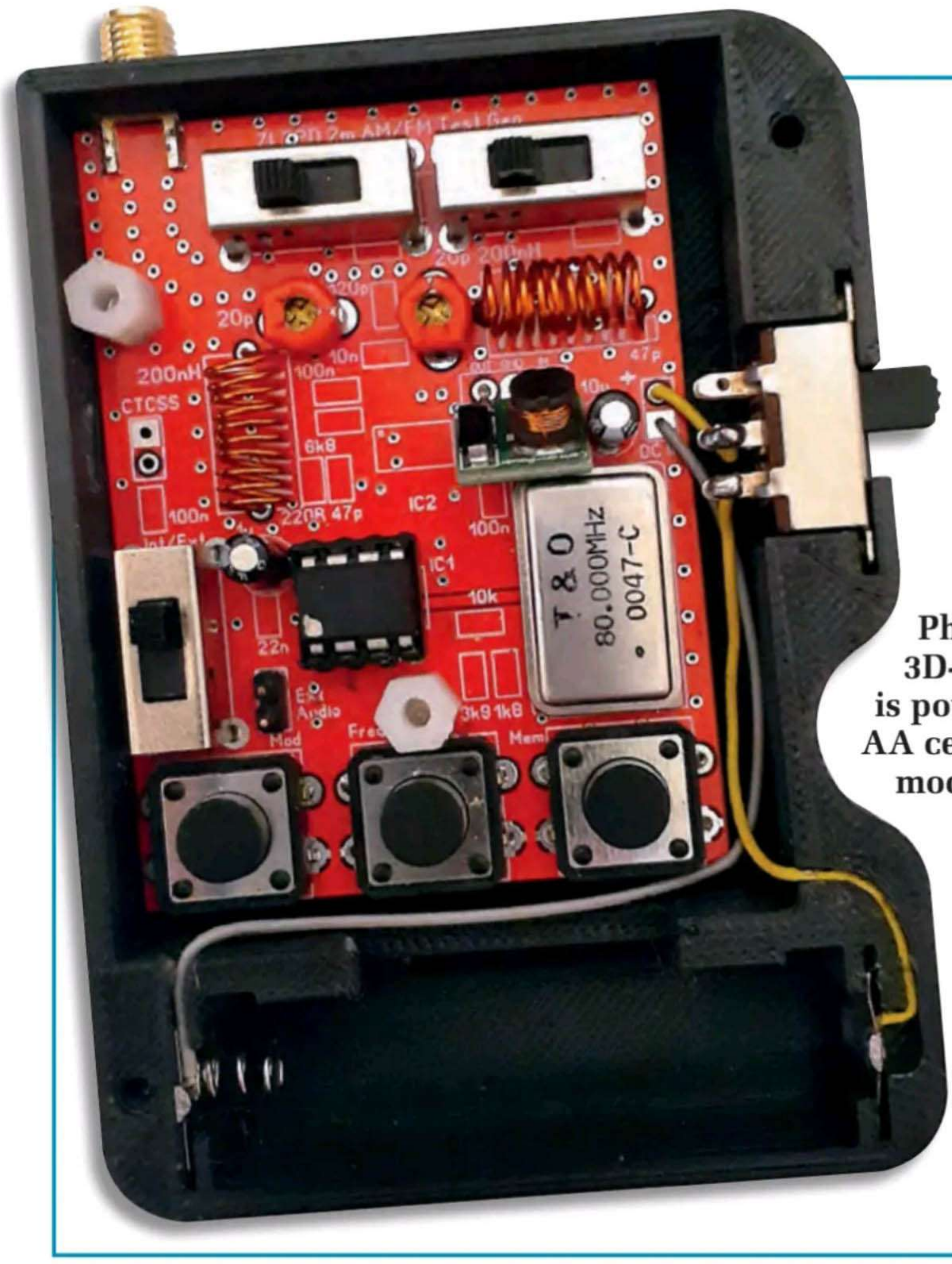


Photo 5: the PCB fits in a 3D-printed enclosure and is powered by a single 1.5V AA cell. The boost regulator module is to the left of the power switch, while the output filter is above and alongside the regulator. The AD9834 and other SMD parts are on the underside of the PCB, but their locations are marked on the component side.

case designed for the board. STL files for the enclosure are available for download along with the software; see pemag.au/Shop/6/266

The 'battery shrapnel' (those metal tabs at each end of the battery compartment) slide into the slots designed for them, and the power switch likewise slots into place in the case.

The PCB can then be mounted. Two 5mm-long, 3mm diameter self-tapping screws pass from the underside into two 6mm-long Nylon spacers to hold the PCB in place. Four more screws hold the front panel in place from the front.

Testing and operation

Insert the programmed ATtiny45 into the socket, ensuring its pin 1 end is at the notched end of the socket; if in doubt, check Fig.7. Switch both attenuator switches on the Test Generator to the left-most positions (minimum attenuation).

Place a 2m VHF FM handheld within about 500mm of the assembled PCB and set it to receive on 146.000MHz. Unmute the receiver so you can hear channel noise and adjust the volume to a suitable level.

Connect and turn on the power to the Test Generator. The handheld's receiver should immediately go quiet, and the handheld's signal strength meter (if there is one) should indicate a very high level (S9 or better, typically). If this does not occur, turn off the power and carefully check your construction.

Assuming your Test Generator passed this test, briefly press and release the Modulation button. You should hear a clean 1kHz tone in the FM handheld's speaker audio.

While monitoring the audio and observing the handheld's signal level meter, adjust the two trimmer capacitors on the Test Generator to achieve the maximum signal level. You might need to move the handheld several metres away (or more) so you begin to hear a little noise on the received signal.

A slight improvement in tuning can sometimes be achieved by very slightly compressing or stretching one or both of the coils, but this is seldom necessary.

The generator starts with the output set to 146.0MHz without modulation. Each press of the Frequency button increments the frequency by 500kHz. If the current frequency is 148.0MHz, the next press will change the output to 144.0MHz, and the 500kHz frequency steps resume again through the 2m band.

Briefly press the Frequency button once to change the Test Generator to 146.5MHz. Retune the handheld's receiver frequency to 146.5MHz; the modulated signal should be audible on this channel.

Press the Modulation button briefly to verify that the modulation can be turned on and off as desired. Ensure modulation is on again before proceeding to the next test.

If the Memory button is pressed, the generator will deliver one of four

What about phase modulation?

You might have noticed I wrote about phase modulation, but the design only supports CW or FM. So what's going on?

Phase and frequency modulation are two sides of the same 'angle modulation' coin. If a modulator is a true phase modulator, the input audio signal is integrated before modulation to produce FM. Likewise, differentiating the input signal to a genuine frequency modulator results in phase modulation.

Hams (amateur radio operators) all talk of "FM", regardless of how it's generated in their radios. When using analog angle modulation ("FM"), it was also always easier to measure frequency deviation when setting up transmitters and everything associated with them. Nobody ever measured phase shift. So everyone talks about FM and frequency deviation when talking about analog angle modulation.

More importantly for the 2m Test Generator is that, for a single tone, FM and PM are indistinguishable.

Curiously then, but logically in context, we only talk about (and measure) specific instantaneous phase shifts when it comes to data angle modulation. Hence PSK, QPSK and 8PSK, where we see a digital application of relatively large phase step modulation.

You might recall that I mentioned the possibility of AM. Not many folk use AM on 2m, but interestingly, combining AM and PM makes it possible to generate 16-QAM with a modest amount of extra effort. But that is an idea for another day.

programmed frequencies starting with the first memory channel. Pressing the Frequency button will return the oscillator output to the currently selected 500kHz frequency increment.

Change the handheld receiver frequency to the frequency you programmed as Memory Channel 1 by pressing the Memory button briefly. The modulated signal should now appear on this channel.

You can continue to press the Frequency or Memory buttons to select and test the other Generator frequencies.

Important notes

Signals generated on some frequencies can produce spurious in-band and out-of-band products. This is to be expected from such a simple DDS generator. Analog Devices warn about this in the data sheet, too.

These additional signals on the output can be generated by the DDS clock, its harmonics and the mixing of one or more aliases. They are typically at least 25dB (aliases) or 30dB (spurious) below the desired output. That's similar to a few legacy commercial RF signal generators.

Given the absence of anti-aliasing audio filtering, avoid the temptation to feed microphone audio into the external audio input or to add a low-power RF amplifier to the Test Generator. It is not suitable for use as a 2m FM transmitter. While usable for basic testing, it will not meet any regulatory test standards for FM transmitters.

External modulation

You can use an external audio source, such as an audio oscillator, to modulate the Test Generator. This can be applied to the CON2 input on the PCB. Ensure the level is between 0V and the supply voltage (ie, 5V maximum with a 5V DC supply). That means the signal must have a ~2.5V DC bias. Signals beyond that limit can cause damage. Monitor the output on a nearby 2m handheld transceiver or receiver to confirm that the modulating tone can be heard.

If you want to add a socket to feed in an external modulation signal, we suggest you couple the signal from that socket to CON2 via a 100nF capacitor and connect 100kΩ resistors from the CON2 signal pin to 5V and GND points on the PCB to get the correct biasing.

CTCSS operation

A suitable CTCSS encoder can also be connected to the CON4 CTCSS input on the PCB. This may be used, for example, to test an FM receiver's CTCSS decoder.

To do this, set the handheld to a suitable channel and set the handheld CTCSS decoder to an appropriate CTCSS frequency, eg, 123.0Hz. Ensure the CTCSS encoder's maximum output level is no more than about 0.5V peak-to-peak.

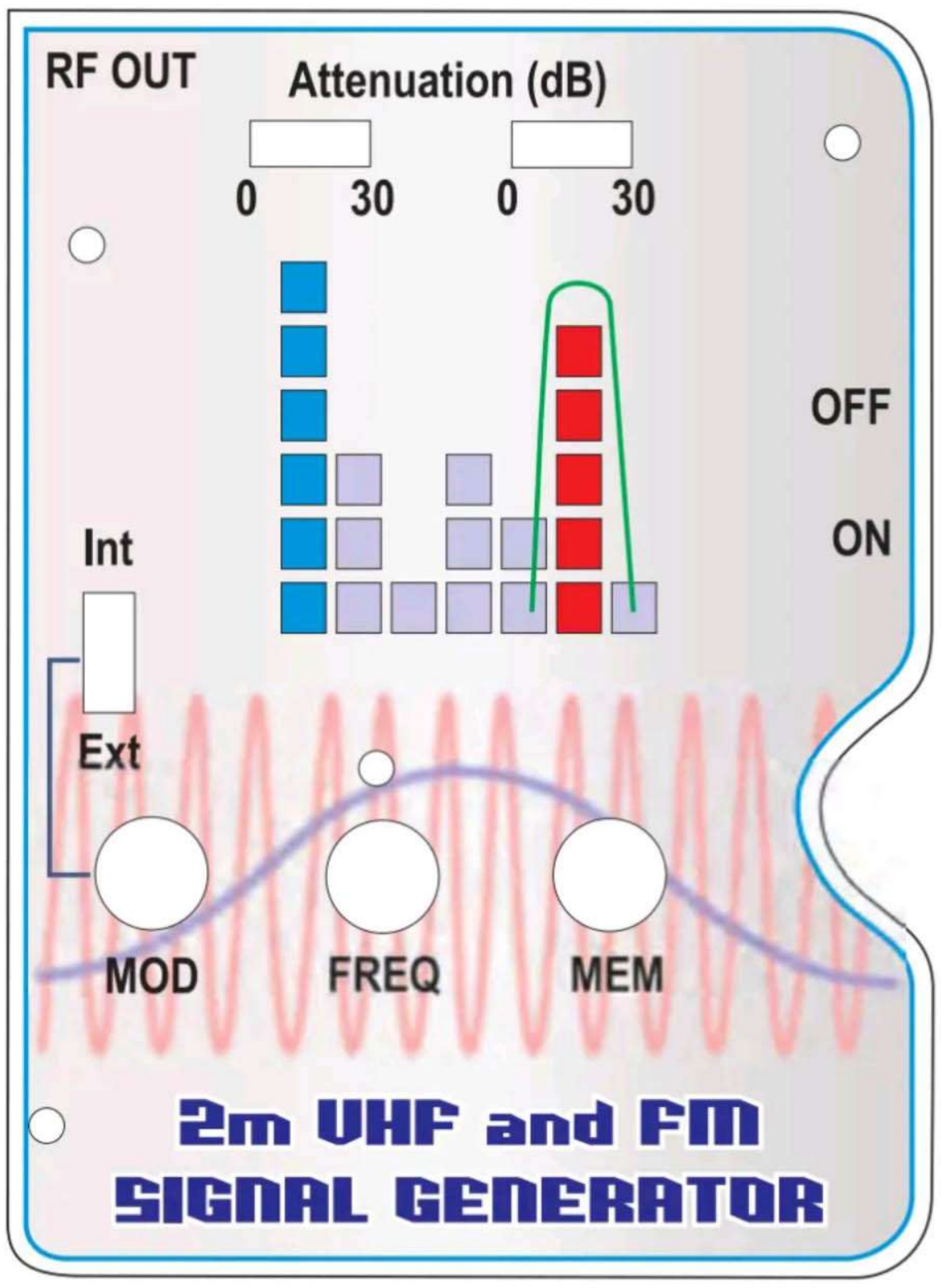


Fig.8: the front panel graphics can be printed and laminated, then glued to the front of the case.

Select internal modulation as the source on the Test Generator, and push S1 briefly to turn on the FM modulation. When the external CTCSS encoder is operating, the receiver's CTCSS decoder should detect the CTCSS tone, unmute the receiver, and the Test Generator's 1kHz tone should be heard.

Turning the external CTCSS encoder off or reducing its level below about 0.1V peak-to-peak should cause the receiver's CTCSS decoder to mute the receiver audio.

However, CTCSS decoders can be very sensitive and may continue to detect a valid tone with CTCSS tone input levels of even 5mV peak-to-peak!

Final comments

The difficulty I encountered in finding basic information and application examples on DDS phase modulation surprised me, given that the data sheets heavily promote this feature. However, achieving precise phase (and frequency) modulation levels with the AD9834 ultimately turned out to be relatively simple.

I hope you find the details and design interesting. This compact FM Test Signal Generator is fun to use and a great conversation piece. **PE**

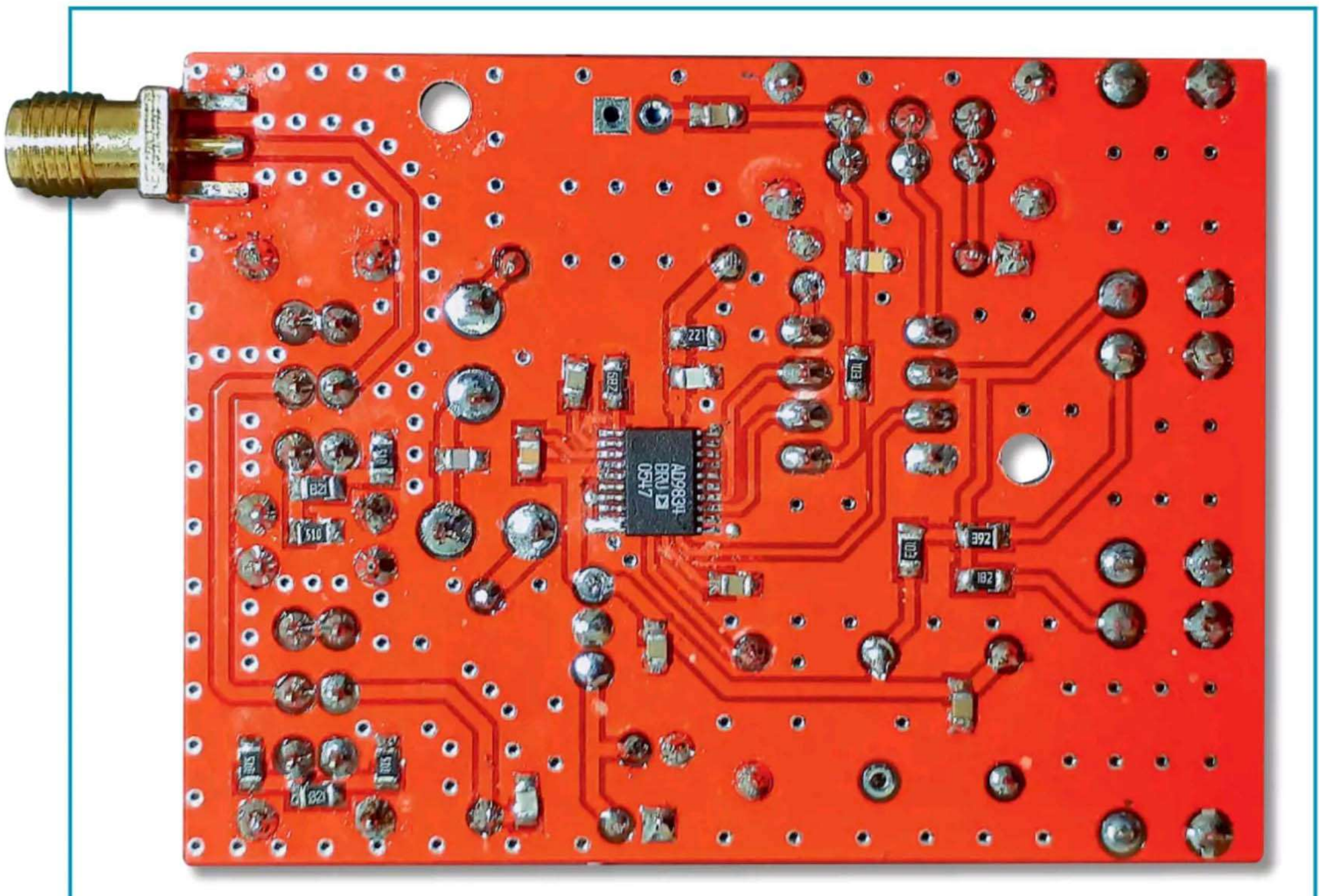


Photo 6: the underside of the PCB with all the SMDs fitted. The bridged pins (9 & 10) on the AD9834 chip are both joined to GND, so I left it like that.

Teach-In 2024

Learn electronics with the ESP32

by Mike Tooley

Part 8 – Where Am I? (GNSS)

In the last part of our *Teach-In* series, we showed how an ESP32 can use an NTP (network time protocol) server to obtain the current time very accurately. Our *Practical Project* was an accurate digital clock displayed on a low-cost 16 × 2 LCD panel, and our *Coding Workshop* introduced arrays and structures.

In this instalment, we explore the use of low-cost GNSS/GPS modules as a means of providing location and time data, possibly for navigation purposes.

The learning objectives for the eighth part of our series are to know how to:

- interface with a low-cost GNSS/GPS module
- decode and interpret data received from GNSS satellites
- use satellite data to determine the precise UTC time
- create user-defined functions

The Global Positioning System (GPS) and GNSS

Originally developed for use by the US military, and known as Navstar (navigation system with timing and ranging), the global positioning system (GPS) was first used in 1978. However, the full constellation of satellites was not in place and fully operational until 1994.

GPS is now widely available for use in many applications including ship, ground vehicle and aircraft navigation.

GPS works by triangulating the distances from several satellites. A minimum of three satellites is required for a GPS receiver to calculate a unique lateral position, while four or more provide additional altitude information (a three-dimensional position fix).

GPS requires a highly accurate time reference. This is provided by atomic clocks in the GPS ground control stations and the satellites themselves. Because of its global accessibility, GPS has now become extremely popular as a source of highly accurate time information.

GPS satellites circle the earth in medium Earth orbit (MEO), at an altitude of approximately 20,200km (12,550 miles). The baseline satellite constellation has 24 satellites in six earth-centred orbital planes. The system can support a constellation of up to thirty satellites in orbit for increased reliability.

The satellites in the GPS constellation are arranged into six equally-spaced orbital planes with four slots each. This ensures that GPS receivers are always within view of at least four baseline satellites.

Individual satellites broadcast their signals on several frequencies in L-band. The three frequencies used by civilian GPS satellites are 1575.42MHz (L1), 1227.60MHz (L2) and 1176.45MHz (L5).

About Teach-In

Our latest *Teach-In* series is about using the popular ESP32 module as a basis for learning electronics and coding. We will be making no assumptions about your coding ability or your previous experience of electronics. If you know one but not the other, you have come to the right place. On the other hand, if you happen to be a complete newbie there's no need to worry because the series will take a progressive hands-on approach. There will be plenty of time to build up your knowledge and plenty of opportunity to test things out along the way.

We've not included too much basic theory because this can be easily found elsewhere, including several of our previous *Teach-In* series, see:

https://bit.ly/pe-ti
https://bit.ly/pe-ti-bundle

Each month, there'll be projects and challenges to help you check and develop your understanding of the topics covered.

To improve the accuracy and resolution, a network of ground-based stations sends correction data to GPS satellites. Such systems include Wide Area Augmentation System (WAAS) in North America and EGNOS (European Geostationary Navigation Overlay Service) in Europe.

The signals transmitted by different GPS satellites use a pseudo-random code comprising a sequence of 1023 binary 1s and 0s. These sequences are modulated using binary phase-shift keying (BPSK).

The resulting modulated signal occupies a significant amount of spectrum space, and a GPS receiver employs a circuit known as a 'correlator' to recover the original narrowband data from the received signal.

In practice, several correlators can be present and operating at the same time. This can significantly reduce the time taken to obtain an initial position fix (TTFF), with different correlators operating on signals from different satellites.

In effect, each of these correlators can be acting on a different information 'channel'. Modern GPS units can use as many as 50 different channels, but typically no more than about a dozen, or so will be in use at any time.

Transient data transmitted by GPS satellites is referred to as "ephemeris data". It provides information about their current (and predicted) location, timing and status. Ephemeris data can also be used to predict future satellite conditions at a specified location and time.

These days, GPS is not the only satellite navigation system. There is also Galileo (EU), GLONASS (Russia), BeiDou (China)

Sentence type	Information content
GPGGA	Time, position and fix type data
GPGLL	Latitude, longitude, UTC time and status
GPGSA	GPS receiver operating mode, number of satellites used in the position solution and DOP values
GPGSV	Number of satellites in view, satellite IDs, elevation, azimuth and signal-to-noise ratio (SNR)
GPMSS	SNR, signal strength, frequency and bit rate from radio-beacon receiver
GPRMC	Time, date, position, course and speed
GPVTG	Course and speed relative to the ground
GPZDA	PPS timing message (synchronized to PPS)

and some regional systems. Many modern 'GPS' receivers can also use some or all of those other systems. With GPS, collectively they are known as Global Navigation Satellite Systems or GNSS.

GNSS data

The data from a GPS/GNSS receiver is a series of ASCII characters, arranged in a format defined by the National Marine Electronics Association (NMEA), so they are referred to as "NMEA sentences".

Several different types of NMEA sentence are produced by receivers. The use of standard ASCII characters makes it possible to receive navigation data using common data interfaces like RS-232, USB, Bluetooth or Wi-Fi. Since not all the received NMEA sentences will be relevant in a particular application, we can ignore those that we are not interested in and concentrate on the sentences containing the data we need.

To do this, we need to recognise the type of sentence before decoding the data contained in it. Luckily, we don't need to write our own code to do this, as there are some excellent libraries available to perform this task for us.

For normal positioning application, the two most-used NMEA sentences are GPGGA and GPRMC (see Table 8.1). These carry GNSS fix data and time, date, position, course and speed data respectively. Note that in some more complex GPS equipment GPGGA is replaced by GNGGA which includes navigation data from multiple global navigation satellite sources (often both GPS and GLONASS satellites). A more complete list of NMEA messages is shown in Table 8.1.

Antennas

GNSS receivers require an antenna to collect signals from GNSS satellites. Since the wavelength is relatively short (approximately 20cm), the antennas can be quite compact. Some GNSS patch antennas can be as small as 1.5×1.5cm, making them ideal for situations where the antenna is mounted on a flat surface, like the windscreen, roof or dashboard of a car.

When mounted on a large ground plane (eg, a car roof), patch antennas can exhibit relatively large gain values. This, coupled with their relatively low profile, makes them ideal for use in vehicle, aircraft and most portable applications.

Small PCB-mounted chip antennas are popular in low-cost applications where space is paramount. Small helical antennas are also found on some equipment. They are particularly suitable for applications when the orientation is unpredictable or a robust external antenna is needed.

NMEA sentences

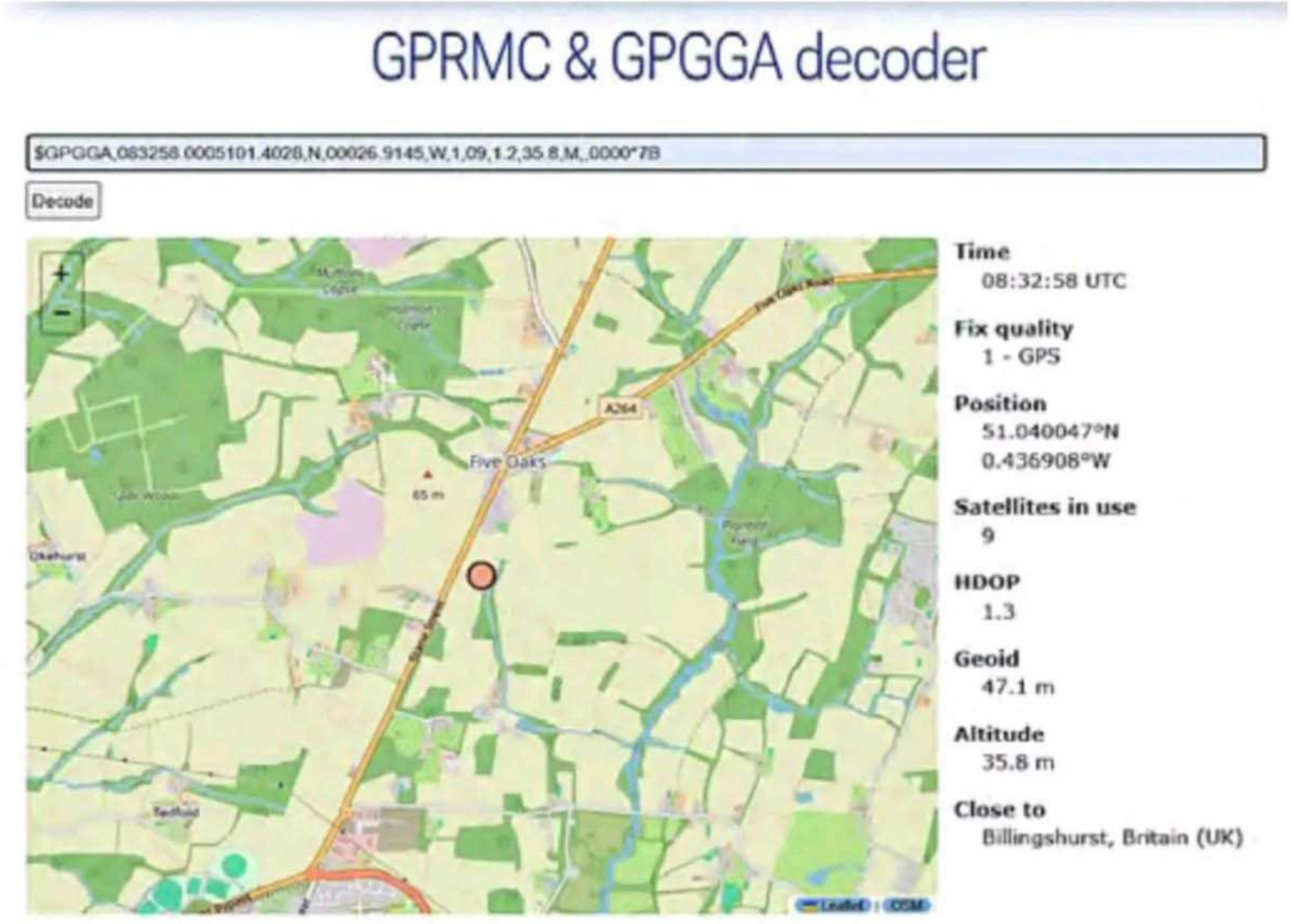
GNSS data is sent as a series of sentences. Here's an example of a typical NMEA GPGGA sentence:

```
$GPGGA,083258.000,5101.4028,N,00026.9145,
        W,1,09,1.3,35.8,M,47.1,M,,0000*7B
```

Let's parse this into some useful data. The first thing we should notice is that the sentence header ($GPGGA) tells us this sentence contains time, position and fix data. From this data we can extract latitude and longitude, as well as a variety of other useful data shown in Table 8.2.

Table 8.2 – Example of the data in a typical NMEA sentence

Name	Value	Format/units	Note
Message ID	$GPGGA		GGA protocol header
Time (UTC)	083258.000	hhmmss.sss	
Latitude	5101.4028	ddmm.mmmm	
N/S Indicator	N	N = north, S = south	
Longitude	00026.9145	ddmm.mmmm	
E/W indicator	W	E = east, W = west	
Position Fix Indicator	1	0=Fix not available or invalid; 1=GPS (SPS mode) fix valid; 2=Differential GPS (SPS mode) fix valid	
Satellites used	9	0 to 12	
HDOP	1.3		Horizontal dilution of precision
Altitude (relative to MSL)	35.8		MSL = mean sea level
Units	M	metres	
Geoid separation	47.1		
Units	M	metres	
Age of diff. corr.	0	second	Only valid when DGPS is used
Diff. ref. Station ID	0000		
Checksum	*7B	Check for errors	
Terminator	<CR><LF>		End of message



GPRMC & GPGGA decoder

$GPGGA,083258.0005101.4028,N,00026.9145,W,1,09,1.2,35.8,M,,0000*7B

Time
08:32:58 UTC

Fix quality
1 - GPS

Position
51.040047°N
0.436908°W

Satellites in use
9

HDOP
1.3

Geoid
47.1 m

Altitude
35.8 m

Close to
Billingshurst, Britain (UK)

Fig.8.1: Robert Larsson's excellent online GPRMC and GPGGA sentence decoder.

You can use an online service to decode the NMEA sentence. The author's favourite online parser is **https://rl.se/gprmc** (see Fig.8.1).

Location fixing and accuracy

A GNSS unit will attempt to fix its location continuously. The rate at which this occurs is referred to as the "update rate". The standard for most GNSS modules is one update per second (ie, an update frequency of 1Hz).

This is adequate for many applications, but faster rates are essential for applications involving aircraft and other fast-moving vehicles. Some modules default to 1Hz but can be configured for, say, 10Hz. Faster update rates will require speedier processing of NMEA sentences.

Several sources of error can affect the GNSS fix. Some errors are introduced by signals received from multiple paths. These can be due to reflections from buildings and other large structures, as well as natural objects such as cliffs and rocky outcrops.

Others can be caused by unwanted ionospheric and tropospheric effects. All these factors can affect the time taken for satellite signals to arrive at a GNSS receiver.

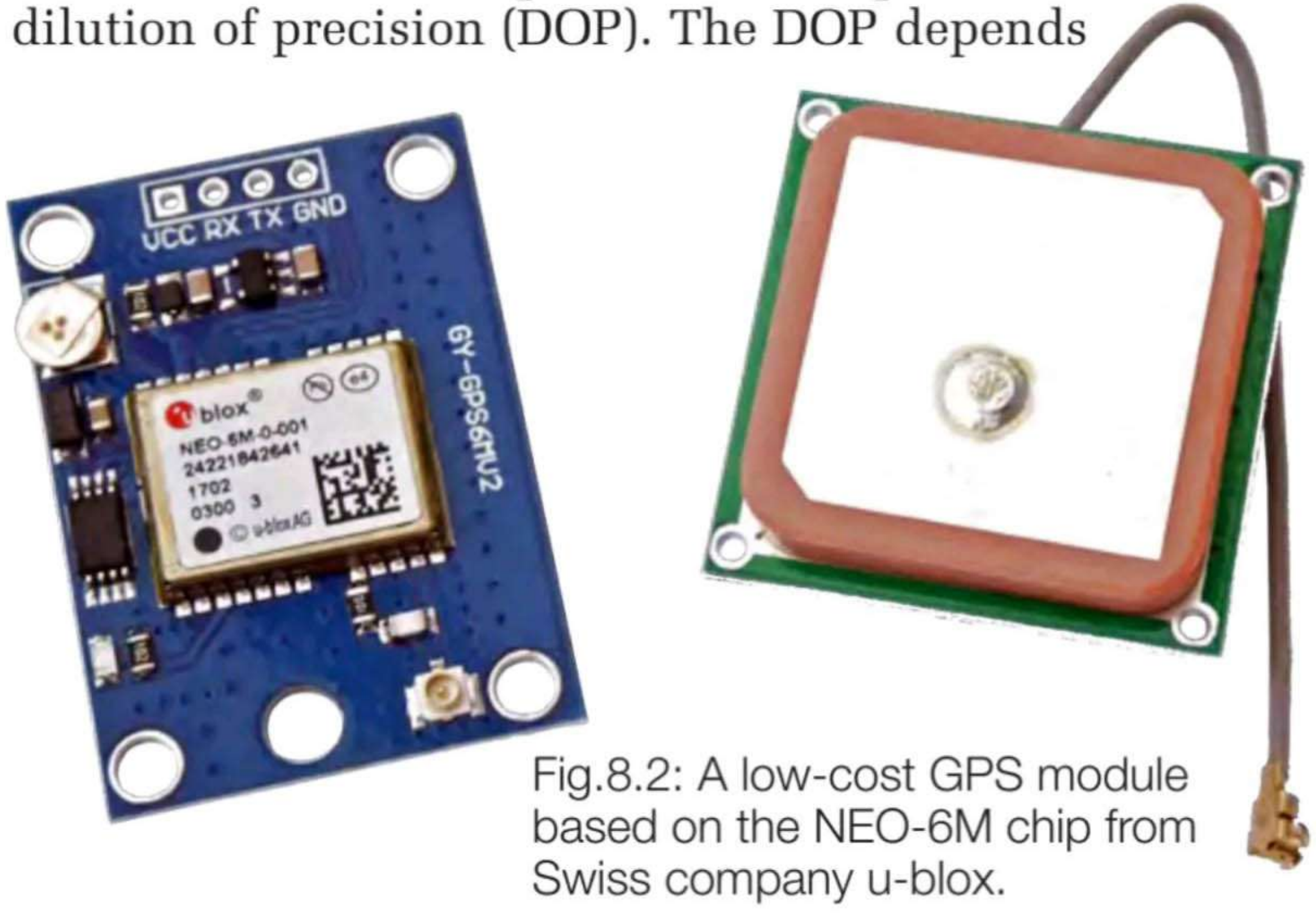


Fig.8.2: A low-cost GPS module based on the NEO-6M chip from Swiss company u-blox.

Gotcha!

The time taken to obtain an initial position (TTFF) while waiting for a GPS module to return data is a key performance indicator of a GPS unit. Since the unit needs to receive signals from as many satellites as possible, the TTFF can be significantly degraded when it is not optimally positioned (eg, indoors) or used in a different location than last time.

Gotcha!

Different startup times are quoted for GPS units depending on whether they are started from cold, warm or hot. A cold start is a restart of a unit after a long period of being powered off. A warm start assumes that ephemeris data is still present and available. A hot start is when the previously received satellite constellation is unchanged from last time, so a fix can be quickly obtained. This is why you sometimes have to wait a while for GPS data to become available.

The combination of errors introduced when the GNSS receiver calculates a position are expressed in terms of dilution of precision (DOP). The DOP depends on the positions of the GNSS satellites relative to the location of the receiver.

Since the position of the satellites is known, it is possible to estimate the accuracy of a GNSS position fix. The DOP calculation uses both the position and number of satellites available. Since Navstar satellites constantly change position their number and position vary, the DOP will also change continuosly.

To make things a little more complicated, the DOP can be divided into vertical and horizontal components (VDOP and HDOP, respectively). Lower values of VDOP and HDOP are associated with more accurate indications. For land- and sea-based operation, HDOP is usually more important.

The conclusion from all of this is that DOP provides you with a valuable indicator of how much confidence you can place on a particular GNSS location fix.

With lower DOP values your results are likely to be much better than when higher values of DOP are prevalent. You can reduce DOP and improve a GNSS unit's positional accuracy by one or more of the following:

1. Placing the antenna of your GNSS receiver (or the complete unit if it has an integrated antenna) so it has a clear and unobstructed view of the full sky. This increases the number of satellites in view and gives the most favourable satellite geometry.
2. Arranging the antenna of your GNSS unit so it is not susceptible to signals reflected from buildings, aircraft and other large structures.
3. Allowing your GNSS to average position over a reasonably long period (eg, several minutes). If you can't, check on the DOP values returned by the GNSS unit, placing more confidence on the data when the lowest values of HDOP (or VDOP) are indicated.

GNSS Modules

The advent of low-cost widely available GNSS receiver modules, such as the one shown in Fig.8.2, has made it possible to add some exciting features to a huge range of microcontroller applications.

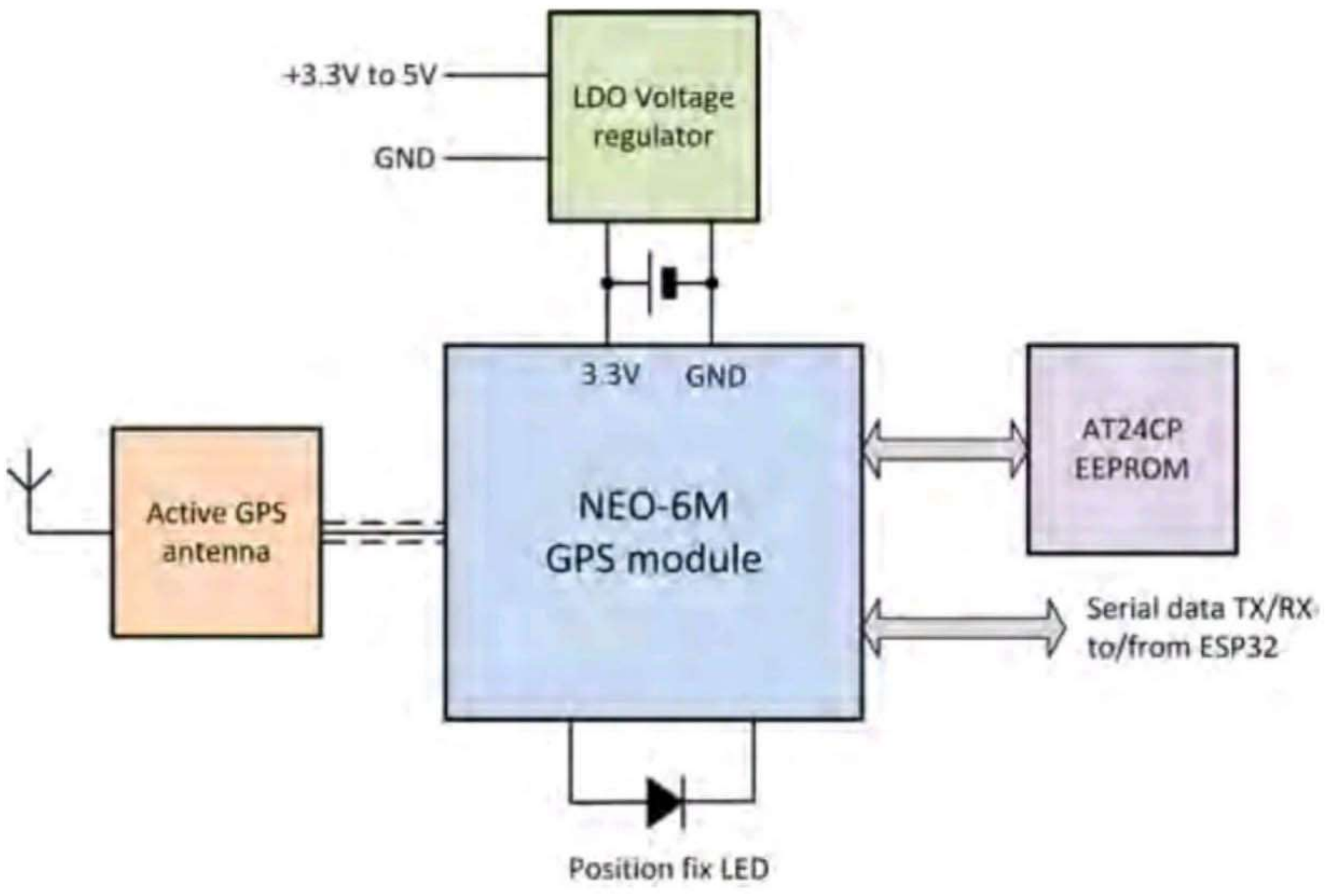


Fig.8.3: A block diagram of the GPS module shown in Fig.8.2.

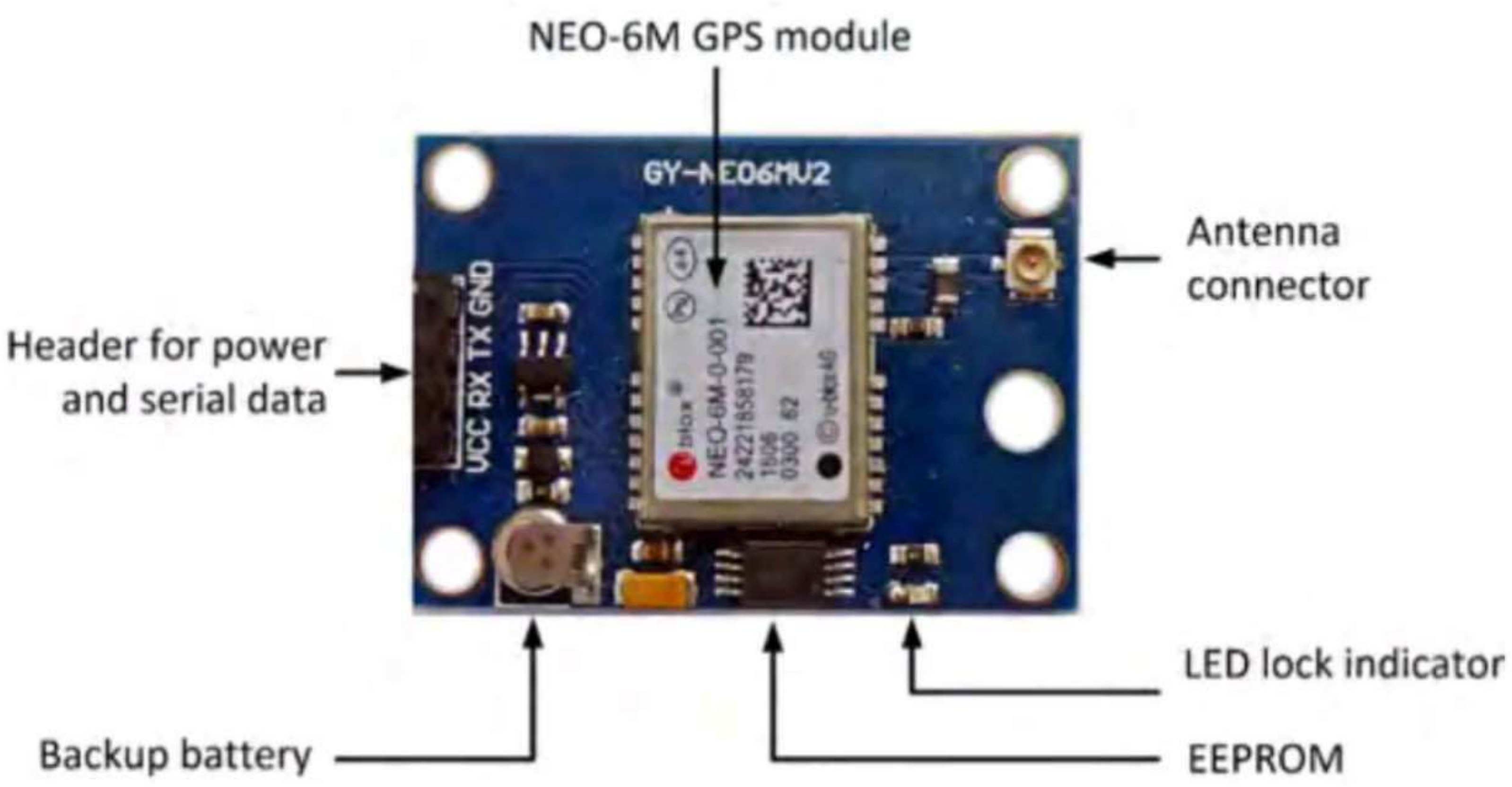


Fig.8.4: The layout of the GPS module board.

For example, they can be used to record accurate position information so that objects can be tracked. Not only is it possible to provide a location fix, but even obtain values for altitude, speed, bearing, distance and time. This can literally add a completely new dimension (or three) to your ESP32 applications!

GNSS modules usually comprise a GNSS receiver chip and GNSS antenna (or a connector for an external active antenna), together with the necessary serial interfacing logic (eg, a UART) and power-conditioning circuitry.

They are available at low cost from suppliers in the UK, USA and Asia. GNSS modules usually operate from a +5V or +3.3V DC supply and employ a serial data interface to transfer information to a computer or microcontroller. This interface usually operates at quite low baud rates, typically 4800 or 9600 bits per second.

The low-cost module shown in Fig.8.2 is based on the popular NEO-6M GNSS receiver chip. It is usually supplied with a matching active antenna measuring a mere 25×25mm. The matching antenna requires mounting on an external surface with a clear and unobstructed view of the sky.

The module incorporates one LED that flashes continuously when a position fix is obtained. It uses a standard TTL serial connection so interfacing with an ESP32 (or another popular microcontroller) is extremely straightforward.

The functional arrangement and board layout for a typical low-cost GNSS module are shown in Fig.8.3 & 8.4.

These modules are usually supplied with a 4-way header that can be soldered in place. Its specifications are shown in Table 8.3.

Table 8.3 – Typical specifications for a low-cost GPS module

Parameter	Specification
Supply voltage range	3.3 to 6V (onboard 3.3V regulator)
Supply current	35mA (11mA in power saving mode)
Maximum I/O voltage	3.6V
Baud rate	4800 to 115,200bps (default 9600)
Operating frequency	1.57542GHz (L1)
Sensitivity	-156dBm (hot start)
Time to first fix (TTFF)	<1s for hot start, ~27s for cold start
Update rate	10Hz maximum (1Hz default)
Number of channels	Up to 50
Positional accuracy	2m (or better) achievable
Velocity accuracy	Typically within 0.1m/s
Maximum velocity	500m/s
Heading accuracy	0.5° achievable whilst in motion
Memory	Battery-backed EEPROM

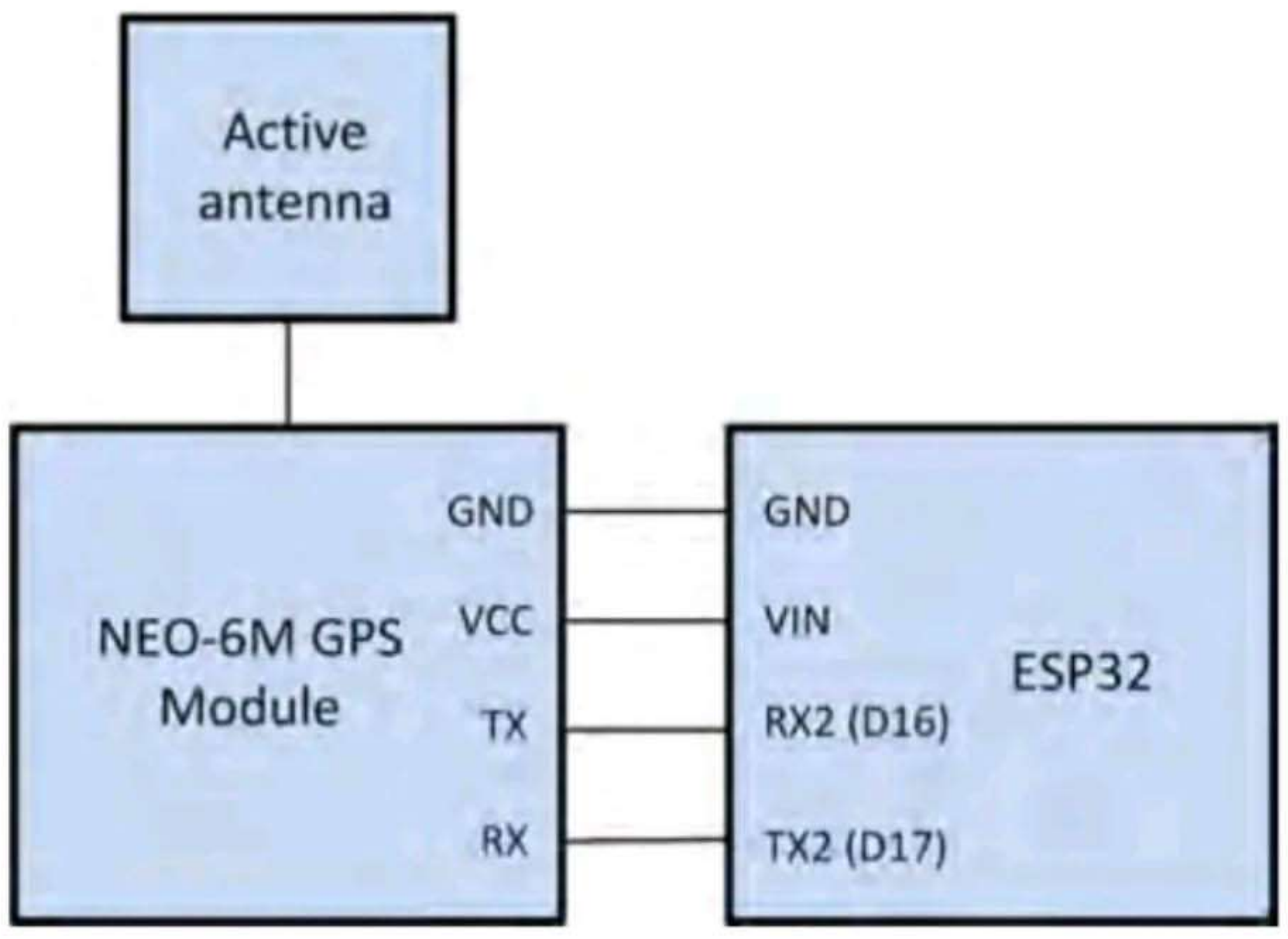


Fig.8.5: Interfacing a low-cost GPS module to an ESP32 development board only requires five wires.

Checking it out

It's time to get to grips with GPS! All you need is the simple four-wire connection from the ESP32 development board to a low-cost GPS module, as shown in Figs.8.5 & 8.6.

When you've completed the wiring, we suggest you jump right in by downloading, compiling and executing Listing 8.1. The code can be downloaded from the PE website by going to https://pemag.au/link/abyr

Listing 8.1 – Using the TinyGPS library to decode NMEA sentences (code extract only)

```
/* Print TinyGPS data using the Serial Monitor. Uses code
developed by Mikal Hart (see TinyGPS documentation) */

#include <SoftwareSerial.h> // EspSoftwareSerial library

#include <TinyGPS.h>     // Mikal Hart's TinyGPS library

TinyGPS gps;
SoftwareSerial ss(16, 17);    // TX and RX serial pins

static void smartdelay(unsigned long ms);
static void print_float(float val, float invalid,
                        int len, int prec);
static void print_int(unsigned long val,
                     unsigned long invalid, int len);
static void print_date(TinyGPS &gps);
static void print_str(const char *str, int len);

void setup()
{
// The setup code is here
}

void loop()
{
// The code for the main loop is here
}
```

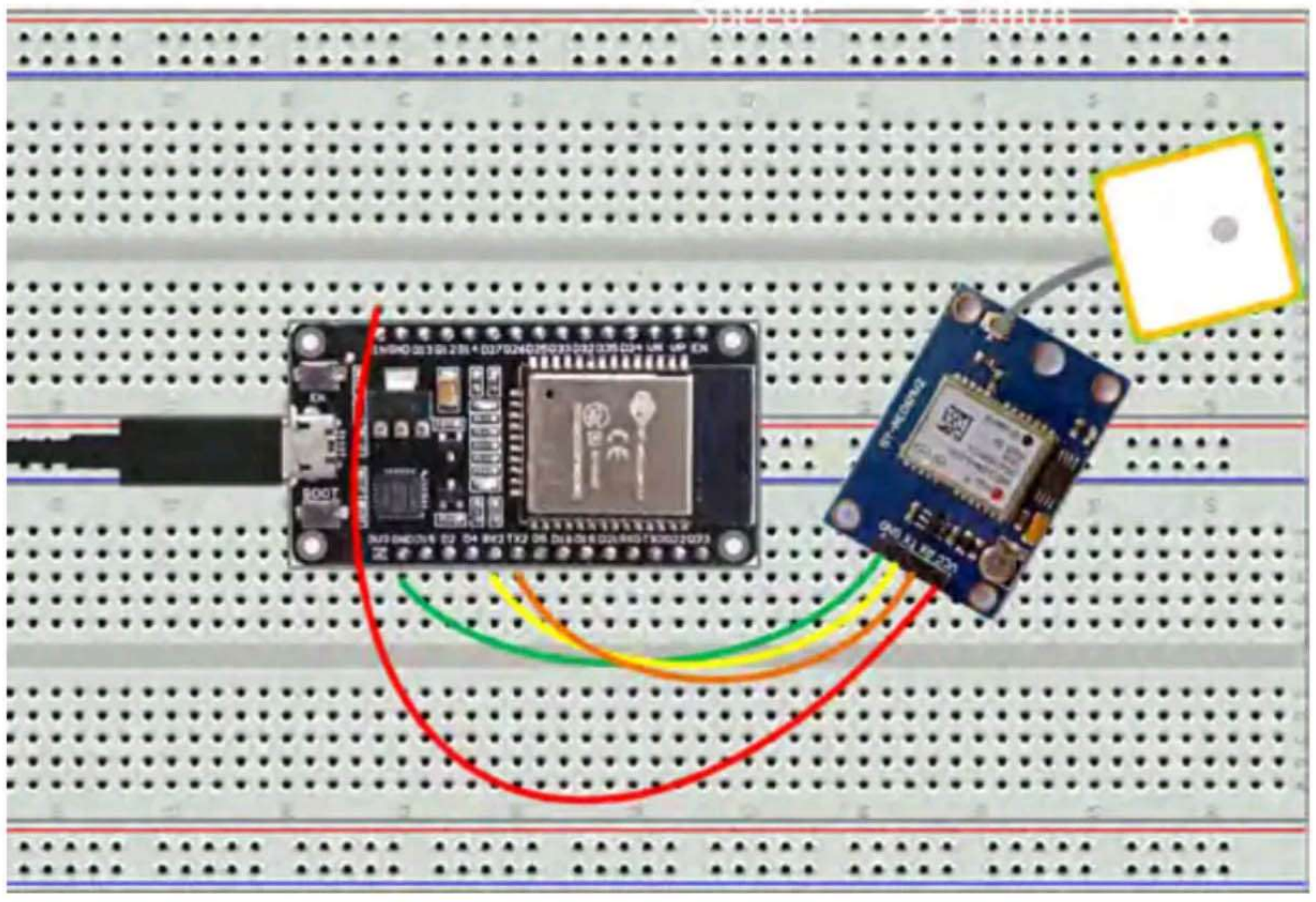
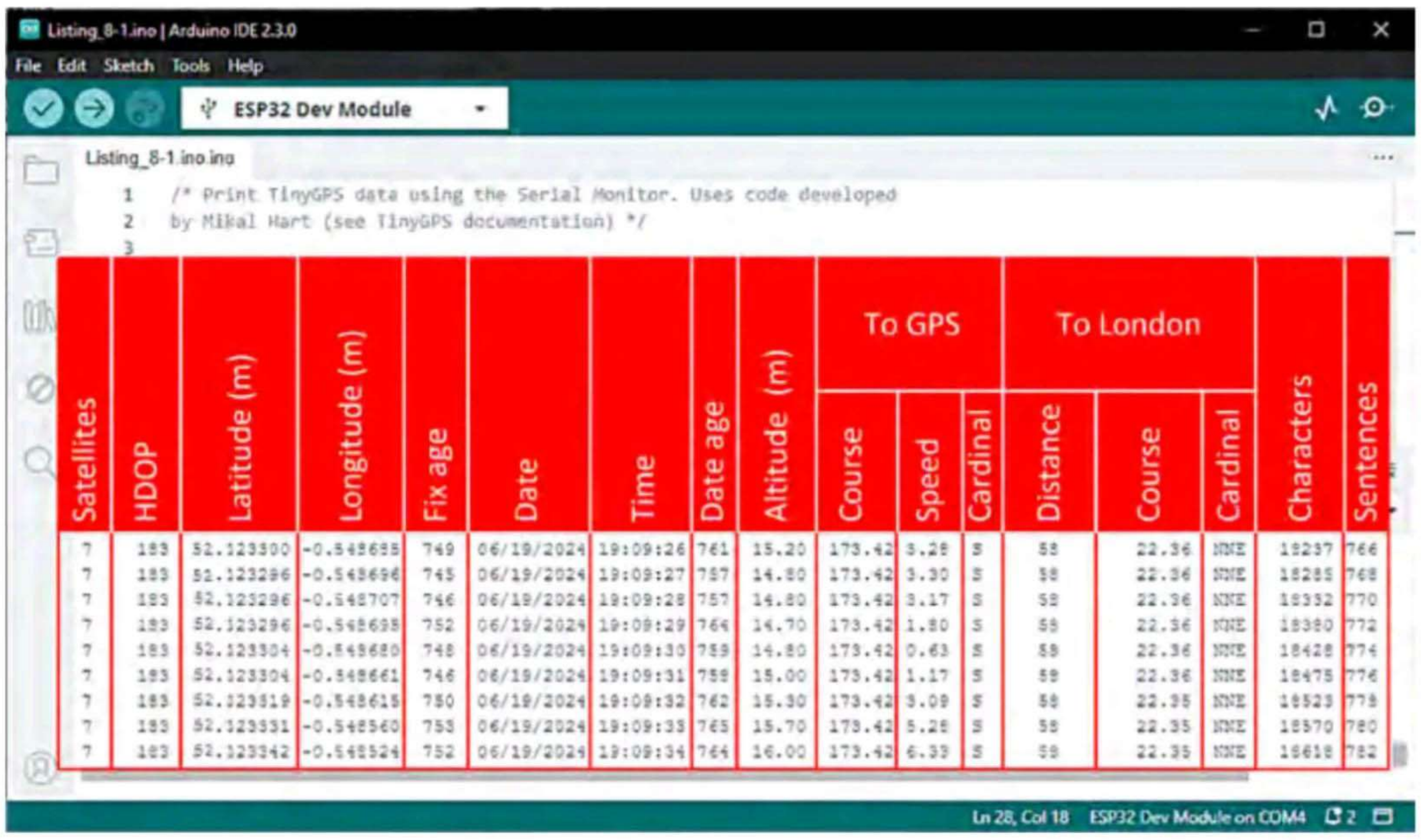


Fig.8.6: A breadboard layout matching Fig.8.5.


Listing_8-1.ino | Arduino IDE 2.3.0
File Edit Sketch Tools Help
ESP32 Dev Module
Listing_8-1.ino.ino
Ln 28, Col 18 ESP32 Dev Module on COM4 2

```
/* Print TinyGPS data using the Serial Monitor. Uses code developed
by Mikal Hart (see TinyGPS documentation) */
```

Satellites	HDOP	Latitude (m)	Longitude (m)	Fix age	Date	Time	Date age	Altitude (m)	To GPS			To London			Characters	Sentences
									Course	Speed	Cardinal	Distance	Course	Cardinal		
7	183	52.123300	-0.548685	749	06/19/2024	19:09:26	761	15.20	173.42	3.28	S	58	22.36	NNE	18237	766
7	183	52.123296	-0.548696	745	06/19/2024	19:09:27	757	14.80	173.42	3.30	S	58	22.36	NNE	18285	768
7	183	52.123296	-0.548707	746	06/19/2024	19:09:28	757	14.80	173.42	3.17	S	58	22.36	NNE	18332	770
7	183	52.123296	-0.548695	752	06/19/2024	19:09:29	764	14.70	173.42	1.80	S	59	22.36	NNE	18380	772
7	183	52.123304	-0.548680	748	06/19/2024	19:09:30	759	14.80	173.42	0.63	S	59	22.36	NNE	18428	774
7	183	52.123304	-0.548661	746	06/19/2024	19:09:31	759	15.00	173.42	1.17	S	59	22.36	NNE	18475	776
7	183	52.123319	-0.548615	750	06/19/2024	19:09:32	762	15.30	173.42	3.09	S	59	22.35	NNE	18523	778
7	183	52.123331	-0.548560	753	06/19/2024	19:09:33	765	15.70	173.42	5.28	S	58	22.35	NNE	18570	780
7	183	52.123342	-0.548524	752	06/19/2024	19:09:34	764	16.00	173.42	6.33	S	59	22.35	NNE	18618	782

Fig.8.7: Typical $GPGGA NMEA data in the Serial Monitor, produced by Listing 8.1.

When you execute the code in Listing 8.1, you will find that it generates and continuously updates a table of values like that shown in Fig.8.7. The data is decoded for successive NMEA $GPGGA sentences, and may at first sight appear baffling. To make it easier to understand, we've added columns (in red) and headings to the figure.

Using GPS data

Now here are a few pointers that might help you develop your own ESP32 code to be used with a GPS module. To begin, you might need to include one or more of the following libraries:

```
#include <LCD.h>
#include <LiquidCrystal_I2C.h>
#include <SoftwareSerial.h>
#include <TinyGPS.h>
```

The somewhat tedious problem of converting NMEA format GPS data into variables that you can use in your code is handled very capably by Mikal Hart's excellent TinyGPS and TinyGPS++ libraries.

This useful library package handles the parsing and conversion of NMEA data and also includes several useful methods and functions that allow you to process and present data in different ways. The TinyGPS library can be downloaded and installed using the IDE's Library Manager. To include the TinyGPS library, use:

```
#include <TinyGPS.h>
```

Listing 8.1 creates a TinyGPS object (we've used the rather obvious title, gps) using:

```
TinyGPS gps;
```

While it is possible to create multiple TinyGPS objects, you will probably not need to do that. A single TinyGPS object will suffice for this and most other applications.

The low-cost GPS module uses a serial interface to communicate with the ESP32, so the code includes the ESP SoftwareSerial library. It can also be obtained from the IDE's Library Manager (make sure that you download and install the ESP32 version of this library).

The serial I/O provided by the GPS module connects directly to the RX2 and TX2 pins of the ESP32 development board (see Fig.8.5). These pins correspond to I/O pins D16 and D17, respectively, so those are the pin numbers that need to be referenced in your code:

```
SoftwareSerial ss(16, 17);  // TX and RX serial pins
```

Each byte present in a received NMEA sentence is handled by the TinyGPS encode() method, using SoftwareSerial to obtain data from the GPS module one character at a time. This process continues as long as valid data is available:

```
void loop() {
  while (ss.available())
  {
    int c = ss.read();
    if (gps.encode(c))  // Valid?
    {
      // New GPS data is processed here
    }
  }
}
```

The next few fragments of code will give you an idea of how data from a GPS module can be handled. For further information, refer to the full code for Listing 8.1 or the examples included with the TinyGPS library.

Display the number of satellites being received using the IDE's serial monitor:

```
Serial.print("Satellites: ");
Serial.println(gps.satellites());
```

... or on an LCD screen:

```
lcd.print("Satellites: ");
lcd.print(gps.satellites());
```

Display the HDOP using the serial monitor:

```
Serial.print("HDOP: ");
Serial.println(gps.hdop());
```

Display the HDOP on an LCD screen:

```
lcd.print("HDOP: ");
lcd.print(gps.hdop());
```

Display the number of characters and sentences received using the IDE's serial monitor:

```
// First we need the statistics to display
unsigned long chars = 0;
unsigned short sentences = 0, failed = 0;
gps.stats(&chars, &sentences, &failed);
// Then we send them to the serial monitor
Serial.print("Characters: ");
Serial.println(chars);
Serial.print("Sentences: ");
Serial.println(sentences);
```

Display the number of characters and sentences received on an LCD screen:

Gotcha!

The TX and RX signals of the serial GPS interface must be cross-connected so that TX for the GPS module is taken to RX2 (D16) on the ESP32 while RX on the GPS module is connected to TX2 (D17) on the ESP32.

Gotcha!

When you've uploaded your code and the ESP32 has reset, the LCD panel will remain blank until the GPS module starts reporting data. This may take several minutes from a cold start and up to a minute following a hot start. Once the unit has acquired a position fix, the module will blink the onboard LED and the LCD should show information (see Fig.8.11).

```
// First we need the statistics to display
unsigned long chars = 0;
unsigned short sentences = 0, failed = 0;
gps.stats(&chars, &sentences, &failed);
// Send the # of characters to the first
// line of the LCD display
lcd.setCursor(0,0);
lcd.print("Characters: ");
lcd.print(chars);
// Send the # of sentences to the second
// line of the LCD display
lcd.setCursor(0,1);
lcd.print("Sentences: ");
lcd.print(sentences);
```

Telling the time (again)

In Part 7 of Teach-In, we showed how the ESP32 could be configured for use as an accurate clock using an NTP time server. We'll now look at an alternative technique for obtaining date and time information using GNSS.

Listing 8.2 – Extracting GPS data and time information

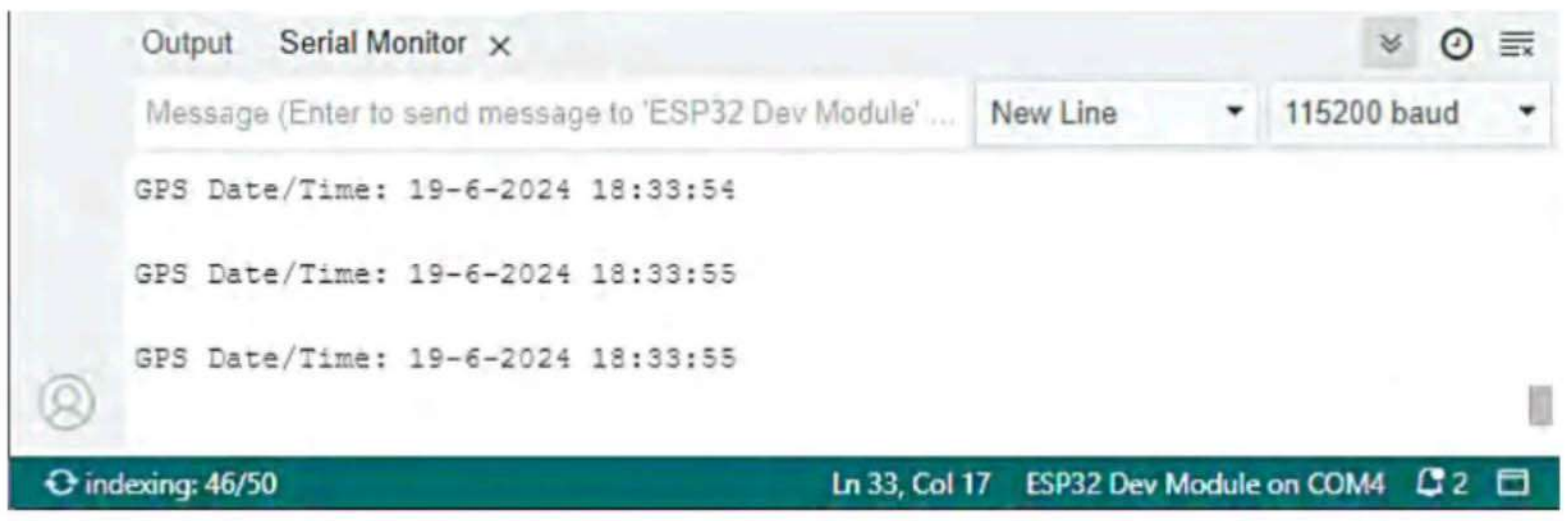
```
/* GPS time/date displayed using the ESP32's Serial Monitor */

#include <TinyGPS++.h>  // The GPS library file
#define GPSbaud 9600    // Set GPS baud rate
TinyGPSPlus gps;        // The GPS object

void setup() {
  // Confirm connection
  Serial.begin(115200);
  Serial2.begin(GPSbaud);
  Serial.println(F("ESP32 GPS data"));
  delay(2000);
}

void loop() {
  if (Serial2.available() > 0) {     // Check serial port
    if (gps.encode(Serial2.read())) { // Read the data
      if (gps.location.isValid()) {   // Check if valid
        if (gps.date.isValid() && gps.time.isValid()) {
          Serial.print(F("GPS Date/Time: "));
          Serial.print(gps.date.day());
          Serial.print(F("-"));
          Serial.print(gps.date.month());
          Serial.print(F("-"));
          Serial.print(gps.date.year());
          Serial.print(F(" "));
          Serial.print(gps.time.hour());
          Serial.print(F(":"));
          Serial.print(gps.time.minute());
          Serial.print(F(":"));
          Serial.println(gps.time.second());
        } else {
          Serial.println(F("Data invalid!"));
        }
        Serial.println();  // Blank line for spacing
      }
    }

    if (millis() > 5000 && gps.charsProcessed() < 10)
      Serial.println(F("Error - GPS data unavailable!"));
  }
}
```



```
GPS Date/Time: 19-6-2024 18:33:54

GPS Date/Time: 19-6-2024 18:33:55

GPS Date/Time: 19-6-2024 18:33:55
```

Fig.8.8: Some date and time data provided by Listing 8.2.

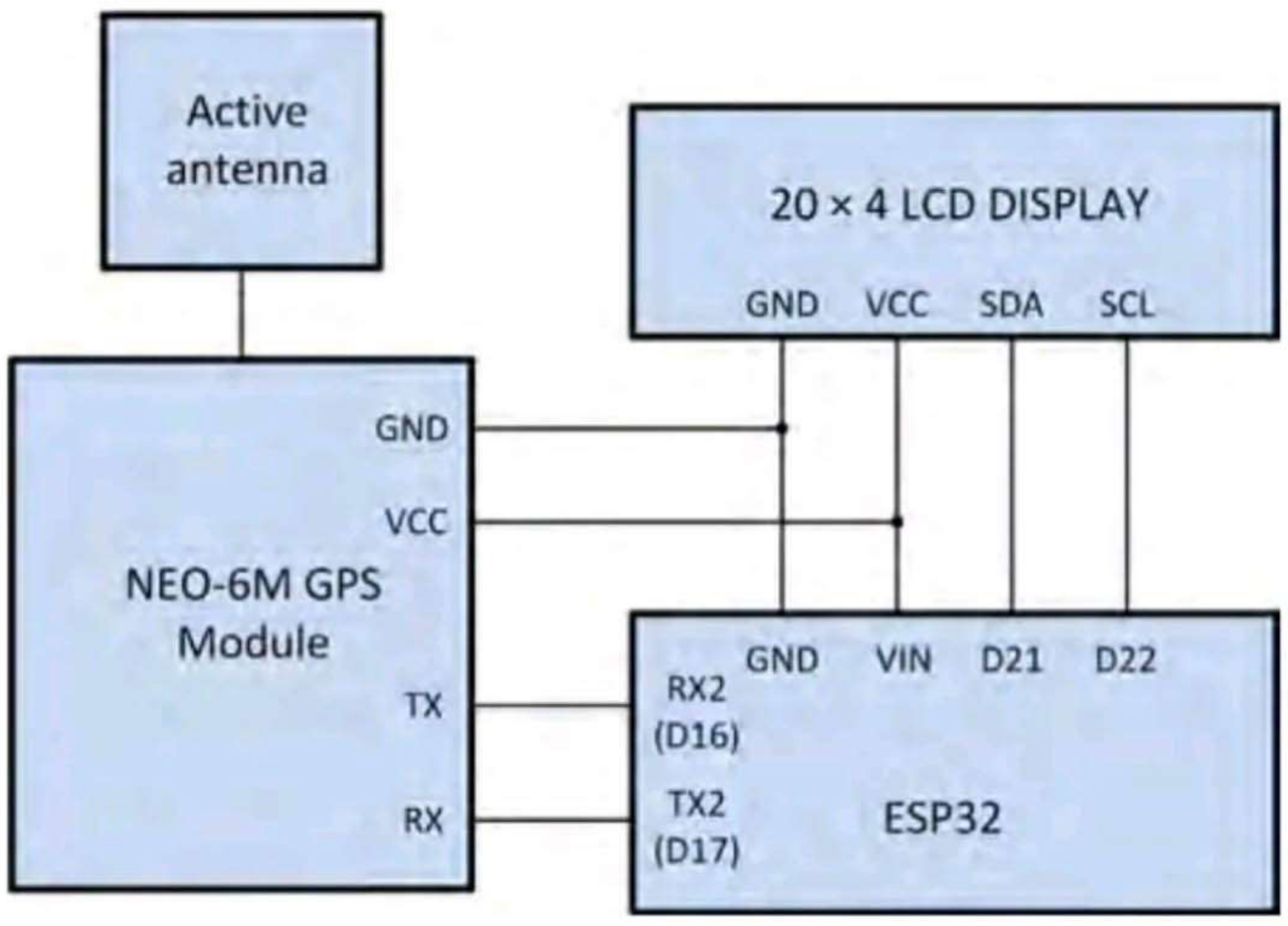


Fig.8.9: The interconnections for the GPS location fixer.

This uses the same interface and wiring arrangement that we described earlier, in Figs.8.5 and 8.6, and the code is shown in Listing 8.2. The output from the Serial Monitor is shown in Fig.8.8.

Practical Project

Our *Practical Project* involves the construction of a simple GPS location fixer using an ESP32 development board, low-cost GPS module, and a 20×4 LCD screen. The required interconnections and wiring are shown in Figs.8.9 and 8.10.

The 20-character, four-line display is like those used previously in this Teach-In series and, as before, it employs an I^{2}C interfaced to minimise wiring and avoid the use of multiple digital I/O lines.

The I^{2}C interface is usually supplied together with the display, and either already soldered in place or comes with a suitable header for interconnection.

The complete code listing for the GPS location fixer is shown in Listing 8.3. The code uses the following libraries: LCD.h, LiquidCrystal_I2C.h, SoftwareSerial.h and TinyGPS.h.

Before finalising your code, you will need to determine the I^{2}C address of your display. Often this will be 0x27 (ie, hexadecimal 27 or decimal 39). You can determine it by running the I^{2}C

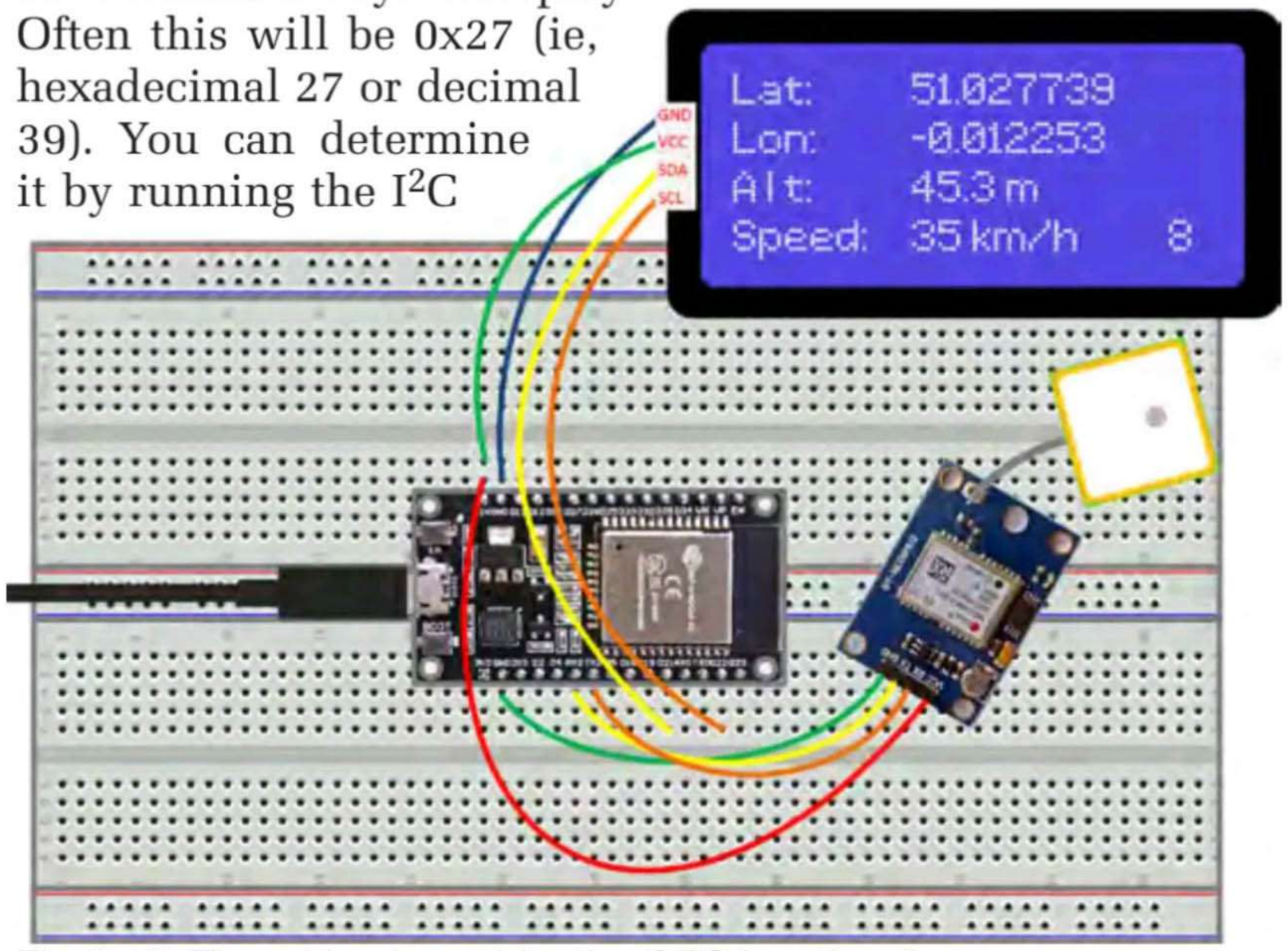


Fig.8.10: The wiring layout for the GPS location fixer.

Scanner application provided in Listing 5.3 (available for download from the PE website at https://pemag.au/link/abys).

Coding workshop

Our *Teach-In* code examples and listings have made liberal use of functions, so you will know that a function is simply a block of code that's been given a name and incorporates statements that will be executed when the function is called.

Functions are a great way to perform repetitive tasks. They help to minimise and simplify your code, making it easier to understand and re-use.

All of our full code listings have included the two most fundamental functions, setup() and loop(). If you've been wondering about the void prefix, it just indicates that the function does not return a value directly.

If a function returns a value, the function definition needs to begin by indicating the type returned (eg, int). The standard format for a function declaration is thus:

```
type functionName(parameters) {
  statement1;
  statement2;
  statement3;
  etc.
  return value;
}
```

Note how the list of function statements is enclosed in curly braces and each statement ends with a semicolon. The parameters that follow a function name enclosed in brackets (parenthese) will be passed to the function. If there are none, the brackets are simply left empty.

Here's an example of a function with a void type that doesn't return anything:

```
void checkLevel() {
  if (tankFull == 1) {
    Serial.println("Warning - tank full!");
  } else {
    Serial.println("Filling, please wait");
  }
}
```

While void functions can still do useful work, you can't use them within expressions, because they don't evaluate to anything.

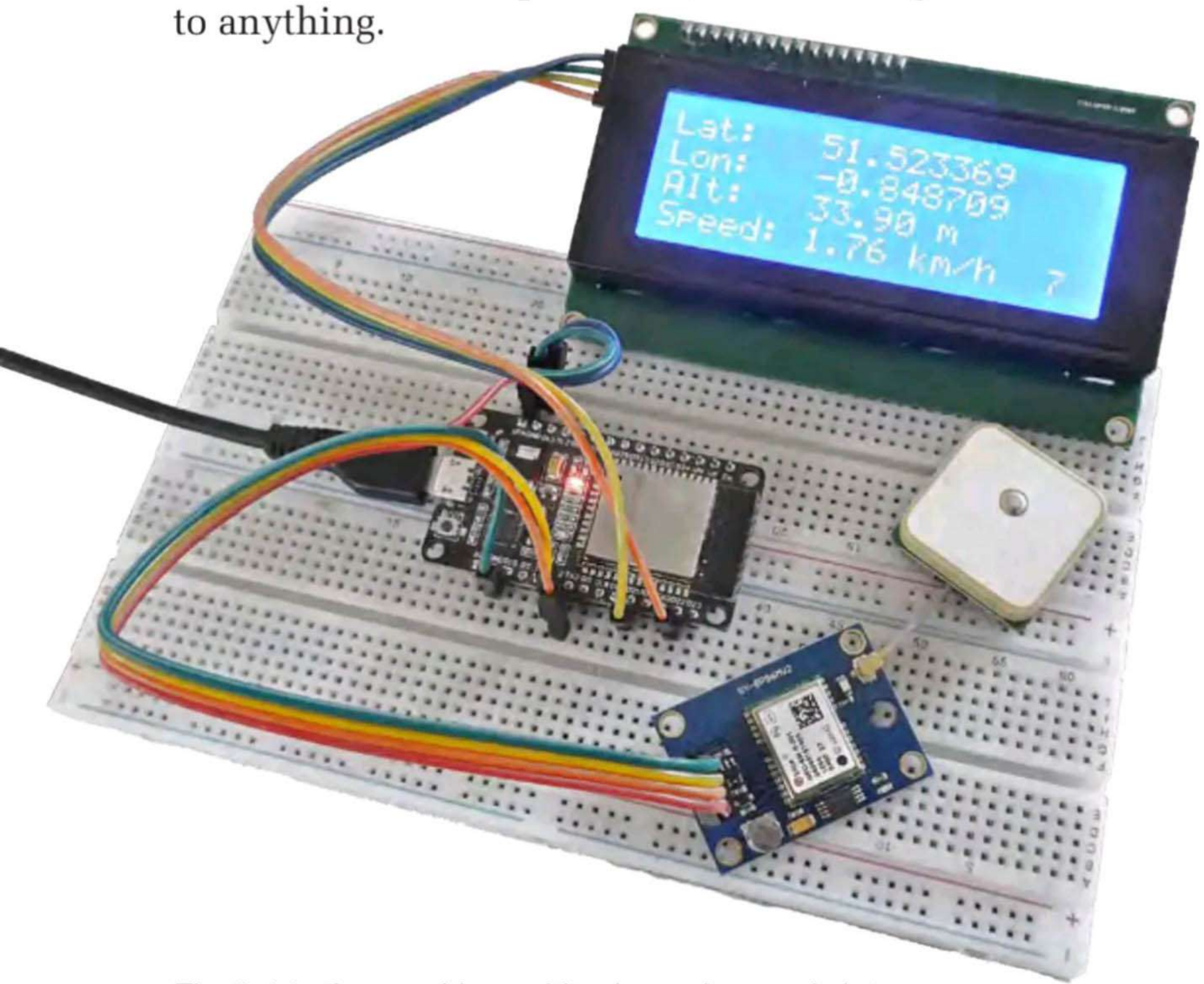


Fig.8.11: the position, altitude and speed data are shown on the LCD panel of the ESP32 GPS location fixer.

Listing 8.3 – Complete code listing for the GPS location fixer

```
/* ESP32 GPS data displayed on an 20x4 LCD panel */

#include <TinyGPS++.h>  // The GPS library file
#define GPSbaud 9600    // Set GPS baud rate
TinyGPSPlus gps;        // The GPS object
#include <LiquidCrystal_I2C.h>
int lcdColumns = 20;
int lcdRows = 4;
LiquidCrystal_I2C lcd(0x27, lcdColumns, lcdRows);

void setup() {
  lcd.init();
  lcd.backlight();
  // Confirm connection
  Serial.begin(115200);
  Serial2.begin(GPSbaud);
  Serial.println(F("ESP32 GPS data"));
  delay(2000);
}

void loop() {
  if (Serial2.available() > 0) {      // Check serial port
    if (gps.encode(Serial2.read())) {  // Read the data
      Serial.println("Data read OK ");
      if (gps.location.isValid()) {    // Check if valid
        Serial.println("Data is valid");
        lcd.setCursor(0, 0);
        lcd.print("Lat: ");
        lcd.setCursor(7, 0);
        lcd.print(gps.location.lat(), 6);
        lcd.setCursor(0, 1);
        lcd.print("Lon: ");
        lcd.setCursor(7, 1);
        lcd.print(gps.location.lng(), 6);
        lcd.setCursor(0, 2);
        lcd.print("Alt: ");
        if (gps.altitude.isValid()) {
          lcd.setCursor(7, 2);
          lcd.print(gps.altitude.meters());
          lcd.setCursor(12, 2);
          lcd.print(" m");
        }
        lcd.setCursor(0, 3);
        lcd.print("Speed: ");
        if (gps.speed.isValid()) {
          lcd.setCursor(7, 3);
          lcd.print(gps.speed.kmph());
          lcd.setCursor(11, 3);
          lcd.print(" km/h");
        }
        lcd.setCursor(18, 3);
        lcd.print(gps.satellites.value());
        lcd.print("*");

        Serial.print(F("DATE/TIME: "));
        if (gps.date.isValid() && gps.time.isValid()) {
          Serial.print(gps.date.day());
          Serial.print(F("-"));
          Serial.print(gps.date.month());
          Serial.print(F("-"));
          Serial.print(gps.date.year());
          Serial.print(F(" "));
          Serial.print(gps.time.hour());
          Serial.print(F(":"));
          Serial.print(gps.time.minute());
          Serial.print(F(":"));
          Serial.println(gps.time.second());
        } else {
          Serial.println(F("Invalid data"));
        }

        Serial.println();
      } else {
        lcd.setCursor(0, 3);
        lcd.print("Wait!");
      }
    }

    if (millis() > 5000 && gps.charsProcessed() < 10)
   Serial.println(F("No GPS data received: check wiring"));
  }
}
```

Listing 8.4 – Passing parameters to a user-defined function

```
/* Example of passing parameters to a function */

int analogPin = 15;

void setup() {
  // Use the serial monitor to display the results
  Serial.begin(9600);
}

void loop() {
  int rawReading;
  int percent;
  // Get a raw reading from the potentiometer
  rawReading = analogRead(analogPin);
  // Send the raw reading to the function
  percent = getPercent(rawReading);
  // Print the result using the Serial Monitor
  Serial.println(percent);
  delay(2000);
}

int getPercent(int inputValue) {
  int result;
  // Map the value to a percentage
  result = map(inputValue, 0, 4095, 0, 100);
  return result;  // and return the value
}
```

Parameters can be passed into a function and results returned to the main loop. Listing 8.4 provides you with a simple example that shows how this is done. The example displays the setting of a potentiometer that has its slider connected to D15 and its two ends taken to +3.3V and GND.

The raw data read from the analog port is passed to the getPercent() function. It, in turn, determines the percentage of the control setting using the map() function. The resulting value is returned to the main loop where, it is displayed using the Serial Monitor.

Note that although we've used a simple loop in this example, the getPercent() function could be called whenever a new percentage reading is needed.

Teach-In Challenge

Extend the functionality of the basic location fixer so that it displays navigational information relative to a destination location such as your home, place of work, or intended destination.

Gotcha!

Several LCD libraries with similar names exist, so it is important to ensure that you use the correct library for the LCD panel that you use. In many cases, guidance is provided by suppliers, but it is worth double-checking that you are using the correct library for the display that you will be using.

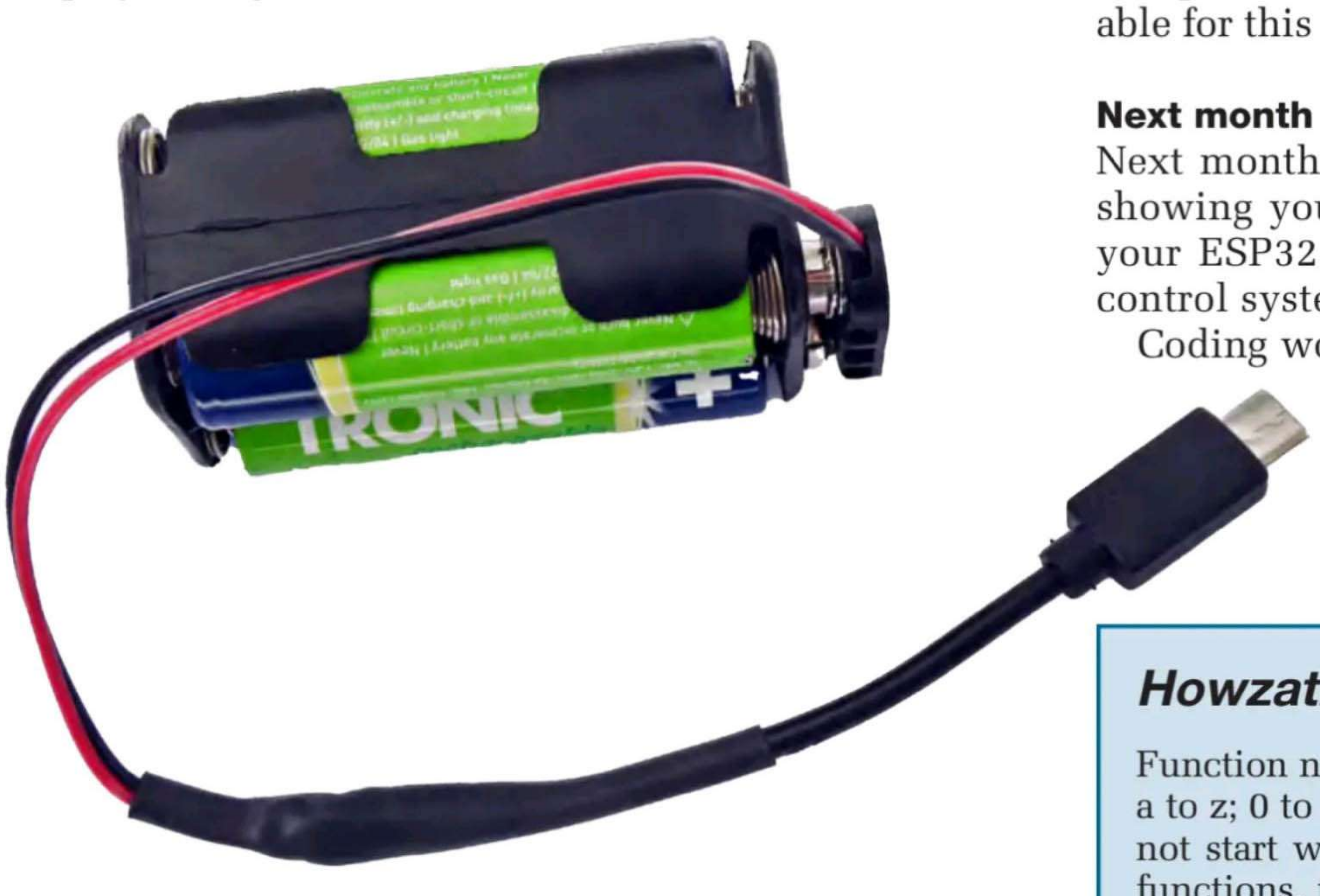


Fig.8.12: a power supply for portable operation using four rechargeable NiMH cells.

The TinyGPS library provides you with methods that will allow you to do this but to get you started here's some code fragments that will get you started.

To get the distance from one location (lat1, long1) to another (lat2, long2), use the following function call:

```
TinyGPS::distance_between(lat1, long1, lat2, long2)
```

It returns the distance in metres between two positions where both are specified as signed decimal degrees of latitude and longitude.

You can get the course from one location (lat1, long1) to another (lat2, long2) using this function call:

```
TinyGPS::course_to(lat1, long1, lat2, long2)
```

It returns the course in degrees (North = 0°, West = 270°) from the first to the second position, where both are specified as signed decimal degrees of latitude and longitude.

Using the above lines of code, if one of the two locations is obtained from gps.get_position() or gps.fget_position() ,it's possible to compute and display the distance and bearing from a fixed point of reference.

The point of reference could be defined as a constant using a line of the form:

```
static const double LONDON_LAT = 51.508131,
                    LONDON_LON = -0.128002;
```

Finally, to carry out tests while moving around, you will need a portable source of USB power. You can easily do this with an ordinary USB 5V "power bank".

Another option is to use four rechargeable NiMH cells mounted in a suitable battery holder (see Fig.8.12). Note that, when series-connected, four standard alkaline cells will provide a higher voltage (6V) and are therefore unsuitable for this use.

Next month

Next month we will be bringing Teach-In to a close by showing you how to interface a matrix keypad and use your ESP32 as the basis of a password-protected entry control system.

Coding workshop will introduce files and file handling techniques that will allow you to save data to an SD or microSD card for later analysis.

Finally, the Checkpoint section will provide you with the ultimate test of your ESP32 knowledge.

Howzat!

Function names can use alphanumeric characters (A to Z; a to z; 0 to 9) and the underscore character, but they must not start with a number. When choosing names for your functions, they must not be the same as the keywords or functions that are already provided by the language.

Practical Electronics PCB SERVICE

PROJECT	CODE	PRICE
OCTOBER 2024		
TQFP-32 Programming Adaptor	24108231	4.99
TQFP-44 Programming Adaptor	24108232	4.99
TQFP-48 Programming Adaptor	24108233	4.99
TQFP-64 Programming Adaptor	24108234	4.99
2m VHF FM Test Signal Generator	06107231	4.99
SEPTEMBER 2024		
Calibrated Microphone (SMD version)	01108231	3.99
Calibrated Microphone (TH version)	01108232	3.99
PIC Programming Adaptor	24105231	4.99
Salad Bowl Speaker crossover	01109231	7.99
AUGUST 2024		
Arduino ESR Meter combined L/C-ESR	04106182	9.99
Arduino ESR Meter separate ESR	04106183	7.99
Watering System Controller	15110231	11.99
JULY 2024		
Dynamic NFC Tag (22 × 31mm)	06101231	5.95
Dynamic NFC Tag (86 × 54mm)	06101232	7.95
Reciprocal Frequency Counter	CSE230101C	8.95
Reciprocal Frequency Counter front panel (black mask)	CSE230102	7.95
Pi Pico-based Thermal Camera	04105231	6.95
JUNE 2024		
Loudspeaker Test Jig	04106231	12.95
MAY 2024		
GPS-Disciplined Oscillator	04103231	12.95
Dual RF Amplifier	CSE220602A	5.95
Songbird	08103231	7.95
APRIL 2024		
Basic RF Signal Generator	CSE221001	8.99
Basic RF Signal Generator front panel (black solder mask)	CSE220902B	9.99
Silicon Chirp Cricket	08101231	7.99
Test Bench 'Swiss Army Knife'	04110221	9.99
MARCH 2024		
Digital Volume Control Potentiometer SMD version	01101231	8.95
Digital Volume Control Potentiometer through-hole version	01101232	9.95
Advanced SMD Test Tweezers body	04106221	6.95
Advanced SMD Test Tweezers legs (pair)	04106212	7.95
FEBRUARY 2024		
Active Mains Soft Starter	10110221	10.95
JANUARY 2024		
Q Meter	CSE220701	8.95
Q Meter front panel (black solder mask)	CSE220704	7.95
Raspberry Pi Pico W BackPack	07101221	8.95
DECEMBER 2023		
Digital Boost Regulator	24110224	8.95
Dual-Channel Power Supply for Breadboards	04112221	11.95
Display Adaptor for the Breadboard PSU	04112222	9.95
NOVEMBER 2023		
LC Meter Mk3	CSE220503C	9.95
DC Supply Filter for vehicles	08108221	8.95
Discrete Audio Op Amp PCB	AO1-JUL23	9.95
Discrete Audio Op Amp PCB + essential components	AO2-JUL23	17.95

PROJECT	CODE	PRICE
OCTOBER 2023		
Buck/Boost Charger Adaptor	14108221	9.95
PIC Breakout Board for SOIC parts	24110225	5.95
PIC Breakout Board for DIP parts	24110222	5.95
AVR64DD32 Breakout board	24110223	5.95
Automatic Train Controller	09109221	6.95
Chuff Sound module	09109222	6.95
SEPTEMBER 2023		
Mini LED Driver	16106221	7.95
New GPS-Synchronised Clock	19109221	9.95
AUGUST 2023		
110dB RF Attenuator	CSE211003	9.95
Universal Battery Charge Controller (2023 update)	14107192	9.95
Wide-Range OhmMeter	04109221	12.95
JULY 2023		
Multimeter Checker/Calibrator	04107221	10.95
MIDI Spectral Sound Synthesiser (full kit – see p.25)	N/A	N/A
JUNE 2023		
Arduino Programmable Load	04105221	8.95
Buck-Boost LED Driver	16103221	8.95
MAY 2023		
Precision AM-FM DDS Signal Generator	CSE211002	14.95
Improved SMD Test Tweezers programmed PIC	0410621PIC	12.95
AO 2x-dual-to-quad through-hole (pack of five)	AO1-MAY23	6.95
AO 2x-dual-to-quad SMD (pack of five)	AO2-MAY23	6.95
AO 2x-single-to-dual through-hole (pack of five)	AO3-MAY23	6.95
AO 2x-single-to-dual SMD (pack of five)	AO4-MAY23	6.95
APRIL 2023		
500W Amplifier Module	see p.22, April 2023	
Clipping Indicator (per channel)	01112211	7.95
CD Welder Power Supply (one needed)	29103221	9.95
CD Welder Controller (one needed)	29103222	9.95
CD Welder Energy Storage module (several needed)	29103223	7.95
AO Universal Dual Op Amp Board	AO1-APR23	9.95
AO Stereo RIAA precision passives kit for dual op amp	AO2-APR23	8.95
MARCH 2023		
Pico BackPack	07101221	9.95
Semaphore Signal (controller)	09103221	7.95
Semaphore Signal (blade)	09103222	5.95
FEBRUARY 2023		
Tesla Coil driver board	26102221	9.95
Tesla Coil potentiometer board	26102222	5.95
Cooling Fan Controller & Loudspeaker Protector	01102221	8.95
Remote Gate Controller	11009121	12.95
JANUARY 2023		
Classic LED Metronome – 8-LED	23111211	7.95
Classic LED Metronome – 10-LED	23111212	8.95
Multi-Channel Speaker Protector – 6 channel	01101221	9.95
Multi-Channel Speaker Protector – 4 channel	01101222	7.95
Remote Control Range Extender – IR-to-UHF	15109212	5.95
Remote Control Range Extender – UHF-to-IR	15109211	7.95
AO Universal Single Op Amp Board	AO1-JAN23	7.95
DECEMBER 2022		
Hummingbird Amplifier	01111211	9.95
SMD Trainer PCB	29106211A	8.95
SMD Trainer PCB + parts	29106211B	13.95

PCBs for most recent *PE/EPE* constructional projects are available. From the July 2013 issue onwards, PCBs with eight-digit codes have silk screen overlays and, where applicable, are double-sided, have plated-through holes, and solder mask. They are similar to photos in the project articles. Earlier PCBs are likely to be more basic and may not include silk screen overlay, be single-sided, lack plated-through holes and solder mask.

Always check price and availability in the latest issue or online. A large number of older boards are listed for ordering on our website.

In most cases we do not supply kits or components for our projects. For older projects it is important to check the availability of all components before purchasing PCBs.

Back issues of articles are available – see Back Issues page for details.

Double-sided | plated-through holes | solder mask

PROJECT	CODE	PRICE
NOVEMBER 2022		
USB Cable tester – main PCB	04108211	12.95
USB Cable tester – front panel	04108212	5.95
USB Cable tester – optional panel	SC5970	5.95
Model Railway Carriage Lights – PCB	09109211	6.95
AO transfmr PCB – standard VTX-A range	VTX-101-007	6.95
AO transfmr PCB – dual-outline VTX102-3001/101-3002	VTX-Dual	6.95
OCTOBER 2022		
SMD Test Tweezers – PCB and pair of tweezer arms	04106211-2	11.95
SMD Test Tweezers – programmed PIC12F1572-I/SN	0410621A	7.95
Tele-com	12110211	12.95
SEPTEMBER 2022		
Touchscreen Digital Preamp – main board	01103191	12.95
Touchscreen Digital Preamp – adaptor board pair	01103192	5.95
20A DC Motor Speed Controller	11006211	9.95
AUGUST 2022		
Multi-purpose Battery Manager – I/O Expander module	11104212	5.95
Multi-purpose Battery Manager – Switch Module	11104211	8.95
Simple MIDI Music Keyboard (for 8 switches)	23101213	6.95
Nano Pong	08105212	7.95
JULY 2022		
Silicon Labs AM/FM/SW Radio	CSE210301C	10.95
Level Crossing Controller	09108211	6.95
JUNE 2022		
Full-wave Motor Speed Controller	1010221	8.95
PIC Programming Helper for 8-pin PICs only	24106211	7.95
PIC Programming Helper for 8, 14 or 20-pin PICs	24106212	10.95
Advanced GPS Computer	05102211	9.95
MAY 2022		
Bus board PCB for Analogue Vocoder	AO1-MAY22	10.95
Complete set of 14 PCBs for Analogue Vocoder	AO2-MAY22	97.95
Programmed EEPROM for Digital FX Unit	FX1-MAY22	10.95
Programmed PIC for Digital FX Unit using potentiometer	FX2-MAY22	8.95
APRIL 2022		
64-key MIDI Matrix shield	23101211	8.95
64-key MIDI Matrix switch board	23101212	11.95
High-current Battery Balancer	14102211	10.95
Digital FX Unit – using potentiometer	01102211	9.95
Digital FX Unit – using BCD switch	01102212	9.95
Universal Audio PSU	AO1-APR22	11.95
MARCH 2022		
Mini Isolated Serial Link	24102211	£5.95
Busy Loo Indicator	16112201	£5.95
Analogue Vocoder – Band-pass filter board	AO1-MAR22	9.95
Analogue Vocoder – HP/LP filter board	AO2-MAR22	9.95
FEBRUARY 2022		
Arduino-based Power Supply	18106201	9.95
Battery Monitor Logger	11106201	10.95
Electronic Wind Chimes	23011201	10.95
Analogue Vocoder – Driver Amplifier	AO-FEB22	8.95
JANUARY 2022		
Vintage battery Radio Li-ion Power Supply	11111201	9.95
MiniHeart: A Miniature Heartbeat Simulator	01109201	8.95
DECEMBER 2021		
AM/FM/SW Digital Receiver	CSE200902A	13.95
Balanced Input and Attenuator for USB CODEC	01106202	11.95

PROJECT	CODE	PRICE
NOVEMBER 2021		
Dual Battery Lifesaver	11111202	£6.95
OCTOBER 2021		
Mini Wi-Fi LCD BackPack	24106201	£8.95
SEPTEMBER 2021		
USB SuperCodec PCB	01106201	£14.95
Audio DDS Oscillator PCB	01110201	£5.95
Audio DDS Oscillator rotary encoder	01110201-ENC	6.95
Programming Adaptor Board for Audio DDS Oscillator	01110202	£5.95
High-power Ultrasonic Cleaner main PCB	04105201	£14.95
High-power Ultrasonic Cleaner front-panel PCB	04105202	
Night Keeper Lighthouse PCB	08110201	£5.95
AUGUST 2021		
Ol' Timer PCB	19104201	£11.95
Ol' Timer 8x8 RGB LED module using WS2812B	19104201-88	£8.95
Ol' Timer set of acrylic case pieces and spacer	19104201-ACR	£8.75
Ol' Timer DS3231 RTC IC wide SOIC-16	19104201-RTC	£5.95
Wideband Digital RF Power Meter	04106201	£9.75
Switchmode 78xx regulators ***(PACK of 5!)***	18105201	£7.95
Cool Beans SMAD display	CB-AUG21	£11.95

For the many pre-2016 PCBs that we stock please see the PE website: www.electronpublishing.com

PE/EPE PCB SERVICE

Order Code Project Quantity Price

....

....

....

....

....

Name ....

Address ....

....

Tel ....

Email ....

I enclose payment of £ .... (cheque/PO in **£ sterling only**)

payable to: **Practical Electronics**

Card No ....

Valid From .... Expiry Date ....

Card Security No ....

You can also order PCBs by phone, email or via the shop on our website: www.electronpublishing.com

No need to cut your issue – a copy of this form is just as good!

All prices include VAT and UK p&p. Add £4 per *project* for post to Europe; £5 per *project* outside Europe.

Orders and payment should be sent to:

Practical Electronics, Electron Publishing Ltd
113 Lynwood Drive, Merley, Wimborne, Dorset BH21 1UU
Tel 01202 880299 Email: shop@electronpublishing.com
On-line Shop: www.epemag.com

Cheques should be made payable to 'Practical Electronics' **(Payment in £ sterling only)**.

NOTE: Most boards are in stock and sent within seven days of receipt of order, please allow up to 28 days delivery if we need to restock.

Practical Electronics PCB SERVICE

JULY 2021

PROJECT	CODE	PRICE
ATtiny816 Breakout / Dev Board with Capacitive Touch	24110181	£9.75
IR Remote Control Assistant (Jaycar version)	15005201	£8.95
IR Remote Control Assistant (Altronics version)	15005202	£8.95
PIC18F Development Board	PNM-JUL21	£12.95
Microphone Preamplifier	AO-JUL21	£11.95

JUNE 2021

PROJECT	CODE	PRICE
Roadies' Test Signal Generator (surface-mount version)	01005201	£8.95
Roadies' Test Signal Generator (through-hole version)	01005202	£9.95
Touchscreen Wide-range RCL Box (Resistor module)	04104201	£18.95
Touchscreen Wide-range RCL Box (Ind/Cap module)	04104202	
KickStart Part 3 – Gyrator-based Audio Filter	KS3-2021	£7.95

MAY 2021

PROJECT	CODE	PRICE
7-Band Equaliser (Mono)	01104201	£8.95
7-Band Equaliser (Stereo)	01104202	£10.95
Car Altimeter	05105201	£7.95

APRIL 2021

PROJECT	CODE	PRICE
Reflow Oven – DSP Active Crossover (CPU)	01106193	£19.95
Reflow Oven – DSP Active Crossover (Front panel)	01106195	
Reflow Oven – DSP Active Crossover (LCD)	01106196	
Frequency Reference Signal Distributor	CSE200103	£8.95

MARCH 2021

PROJECT	CODE	PRICE
Nutube Guitar Effects Pedal	01102201	£12.95
Programmable Thermal Regulator (Peltier Interface)	21109181	£18.95
Programmable Thermal Regulator (Peltier Driver)	21109182	
Tunable HF Preamp	CSE190502	£8.95

FEBRUARY 2021

PROJECT	CODE	PRICE
4G Remote Monitoring	27111191	£9.95

JANUARY 2021

PROJECT	CODE	PRICE
Nutube Valve Preamplifier	01112191	£12.95
Arduino DCC Controller	09207181	£10.95

DECEMBER 2020

PROJECT	CODE	PRICE
Pseudo-Random Sequence Generator	16106191	£7.95
Clever Charger	14107191	£11.95
PE Theremin Amplifier	AO-1220-01	£8.95

NOVEMBER 2020

PROJECT	CODE	PRICE
LED Christmas Tree (1 off)	16107181-1	£6.95
LED Christmas Tree (4 off)	16107181-2	£14.95
LED Christmas Tree (12 off)	16107181-3	£24.95
LED Christmas Tree (20 off)	16107181-4	£34.95
USB/SPI Interface Board	16107182	£5.95
45V/8A Power Supply PCB *plus* acrylic spacer	18111181	£14.95
45V/8A Power Supply front panel five-way display bezel	18111181-BZ	£3.95
Five-way LCD Panel Meter/Display	18111182	£7.95

OCTOBER 2020

PROJECT	CODE	PRICE
Digital Audio Millivoltmeter	04108191	£9.95
Precision Signal Amplifier	04107191	£7.95

SEPTEMBER 2020

PROJECT	CODE	PRICE
PE Theremin PSU	AO-0920-01	£5.95
PE Theremin PSU transformer	AO-0920-02	£7.95
Micromite Explore-28	07108191	£6.95
Ultrabrite LED Driver	16109191	£6.95

AUGUST 2020

PROJECT	CODE	PRICE
Micromite LCD BackPack V3	07106191	£9.95
Steering Wheel Audio Button to Infrared Adaptor	05105191	£7.95

JULY 2020

PROJECT	CODE	PRICE
AM/FM/CW Scanning HF/VHF RF Signal Generator	04106191	£13.95
Speech Synthesiser with the Raspberry Pi Zero	01106191	£5.95
PE Mini-organ PCB	AO-0720-01	£14.95
PE Mini-organ selected parts	AO-0720-02	£8.95
High-current Solid-state 12V Battery Isolator – control	05106191	£6.95
High-current Solid-state 12V Battery Isolator FET (2oz)	05106192	£9.95

JUNE 2020

PROJECT	CODE	PRICE
Arduino breakout board – 3.5-inch LCD Display	24111181	£6.95
Six-input Audio Selector main board	01110191	£10.95
Six-input Audio Selector switch panel board	01110192	

MAY 2020

PROJECT	CODE	PRICE
Ultra-low-distortion Preamplifier Input Selector	01111112	£11.25
Ultra-low-distortion Preamplifier pushbutton Input Selector	01111113	
Universal Regulator	18103111	£7.95
433MHz Wireless Data Repeater	15004191	£8.50
Bridge-mode Adaptor for Amplifier	01105191	£7.95
iCEstick VGA Terminal	02103191	£5.95
Analogue noise with tilt control	AO-0520-01	£7.95
Audio Spectrum Analyser	PM-0520-01	£8.95

APRIL 2020

PROJECT	CODE	PRICE
Flip-dot Display black coil board	19111181	£14.95
Flip-dot Display black pixels	19111182	
Flip-dot Display black frame	19111183	
Flip-dot Display green driver board	19111184	

MARCH 2020

PROJECT	CODE	PRICE
Diode Curve Plotter	04112181	£10.95
Steam Train Whistle / Diesel Horn Sound Generator	09106181	£8.50
Universal Passive Crossover (one off)	UPC0320	£12.50

FEBRUARY 2020

PROJECT	CODE	PRICE
Motion-Sensing 12V Power Switch	05102191	£5.95
USB Keyboard / Mouse Adaptor	24311181	£8.50
DSP Active Crossover (ADC)	01106191	£29.95
DSP Active Crossover (DAC) ×2	01106192	
DSP Active Crossover (CPU)	01106193	
DSP Active Crossover (Power/routing)	01106194	
DSP Active Crossover (Front panel)	01106195	
DSP Active Crossover (LCD)	01106196	

JANUARY 2020

PROJECT	CODE	PRICE
Isolated Serial Link	24107181	£8.50

DECEMBER 2019

PROJECT	CODE	PRICE
Extremely Sensitive Magnetometer	04101011	£16.75
Four-channel High-current DC Fan and Pump Controller	05108181	£8.75
Useless Box	08111181	£11.50

NOVEMBER 2019

PROJECT	CODE	PRICE
Tinnitus & Insomnia Killer (Jaycar case – see text)	01110181	£9.95
Tinnitus & Insomnia Killer (Altronics case – see text)	01110182	£9.95

OCTOBER 2019

PROJECT	CODE	PRICE
Programmable GPS-synced Frequency Reference	04107181	£11.50
Digital Command Control Programmer for Decoders	09107181	£9.95
Opto-isolated Mains Relay (main board)	10107181	£11.50
Opto-isolated Mains Relay (2 × terminal extension board)	10107182	

AUGUST 2019

PROJECT	CODE	PRICE
Brainwave Monitor	25108181	£12.90
Super Digital Sound Effects Module	01107181	£6.95
Watchdog Alarm	03107181	£8.00
PE Theremin (three boards: pitch, volume, VCA)	PETX0819	£19.50

CLASSIFIED ADVERTISING

Practical Electronics

If you want your advertisements to be seen by the largest readership at the most economical price then our classified page offers excellent value.

The rate for semi-display space is £10 per centimetre high, with a minimum height of 2·5cm. All semi-display advertisements have a width of 5.5cm. The prepaid rate for classified adverts is 40p per word (minimum 12 words).

For rates, further information on display and classified advertising or to book an ad, please email us at **pe@pemag.au**

Practical Electronics reaches more UK readers than any other UK monthly hobby electronics magazine. Our sales figures prove it.

We have been the leading monthly magazine in this market for the last twenty-seven years.

BOWOOD ELECTRONICS LTD
Suppliers of Electronic Components
www.bowood-electronics.co.uk
Unit 10, Boythorpe Business Park, Dock Walk, Chesterfield,
Derbyshire S40 2QR. Sales: 01246 200 222
Send large letter stamp for Catalogue

Electrical Industries Charity (EIC)
We help people working in the electrical, electronics and energy community as well as their family members and retirees.
We use workplace programmes that give the industry access to financial grants and a comprehensive range of free and confidential services.
www.electricalcharity.org
electrical industries charity

COAST ELECTRONICS
BREAKOUTS-COMPONENTS-CONTRACT DESIGN-3D PRINTER PARTS-MUSICAL-MICROCONTROLLERS
WWW.COASTELECTRONICS.CO.UK

Andrew Kenny – Qualified Patent Agent
EPO UKIPO USPTO
Circuits Electric Machinery Mechatronics
Web: www.akennypatentm.com
Email: Enquiries@akennypatentm.com
Tel: 0789 606 9725

MISCELLANEOUS

VALVES AND ALLIED COMPONENTS?
For free stock list and/or advice, please contact me: geoffdavies337@gmail.com Telephone: 01788 574774

PIC DEVELOPMENT KITS, DTMF kits and modules, CTCSS Encoder and Decoder/Display kits.
Visit www.cstech.co.uk

COMPONENTS AS JOB LOT
Large stock. Transistors, Diodes, Logic, Memory and Analogue parts. 1000s of passive items. Catalogued and sorted. 'New Old Stock' in racks.
Email: rojobarn@gmail.com

ADVERTISING INDEX

AOSHOP 35
FLOWCODE . . . . Inner front cover
HAMMOND ELECTRONICS Ltd. 6
MICROCHIP TECHNOLOGY . . . . Outer back cover
TEKTRAIN 41

Advertising mailing address
Electron Publishing Australia
PO Box 194
Matraville NSW 2036
Australia

Email pe@pemag.au
Web www.electronpublishing.com

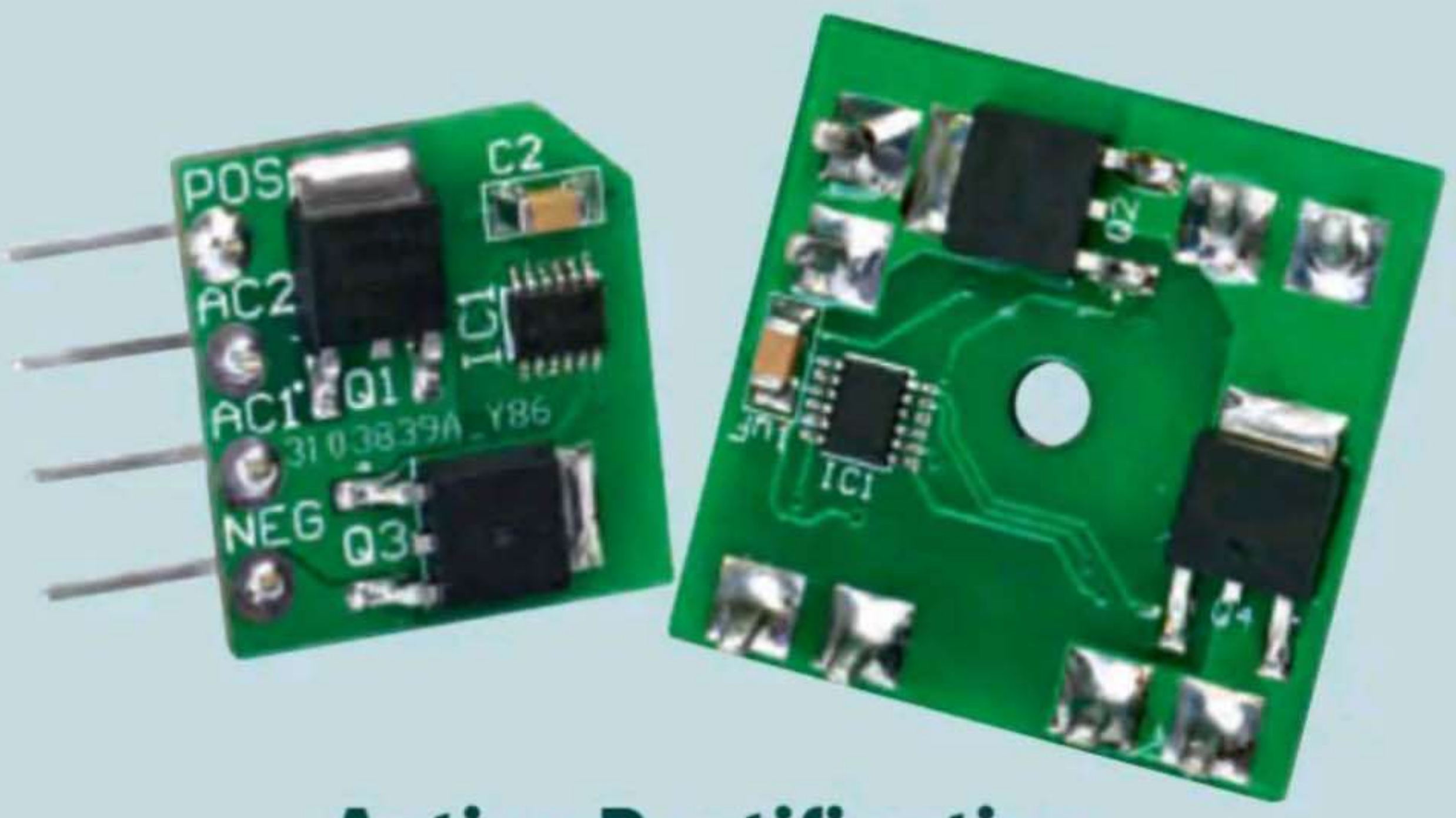
For editorial contact details, see page 2.

Next Month – in the November issue

TIMER
ON
OFF
SECURE REMOTE SWITCH TRANSMITTER

Secure Remote Switch

Relay switching with up to 16 keyfob transmitters and rolling code security



Active Rectification

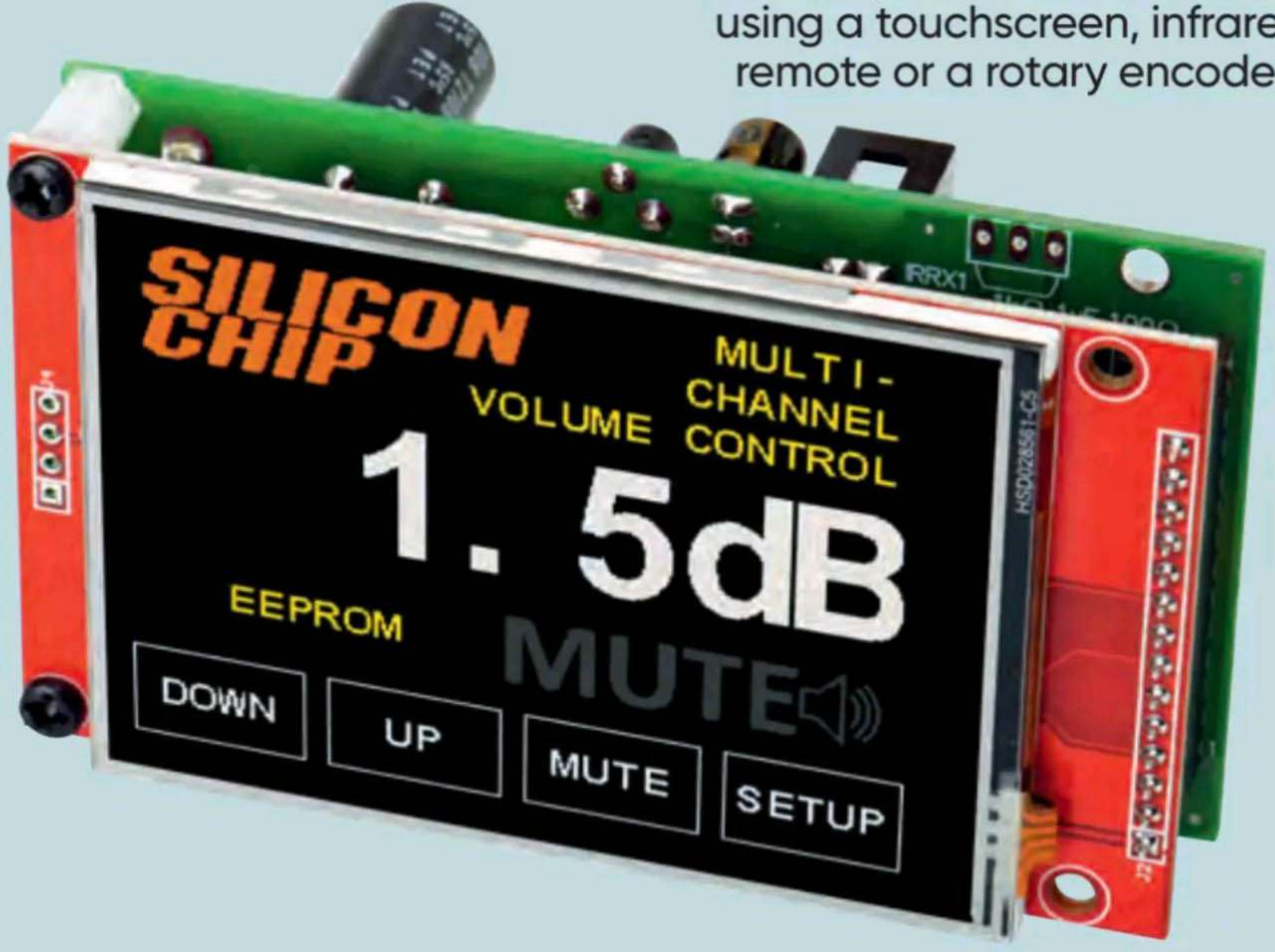
Six different PCBs for highly efficient conversion of low-voltage AC to DC

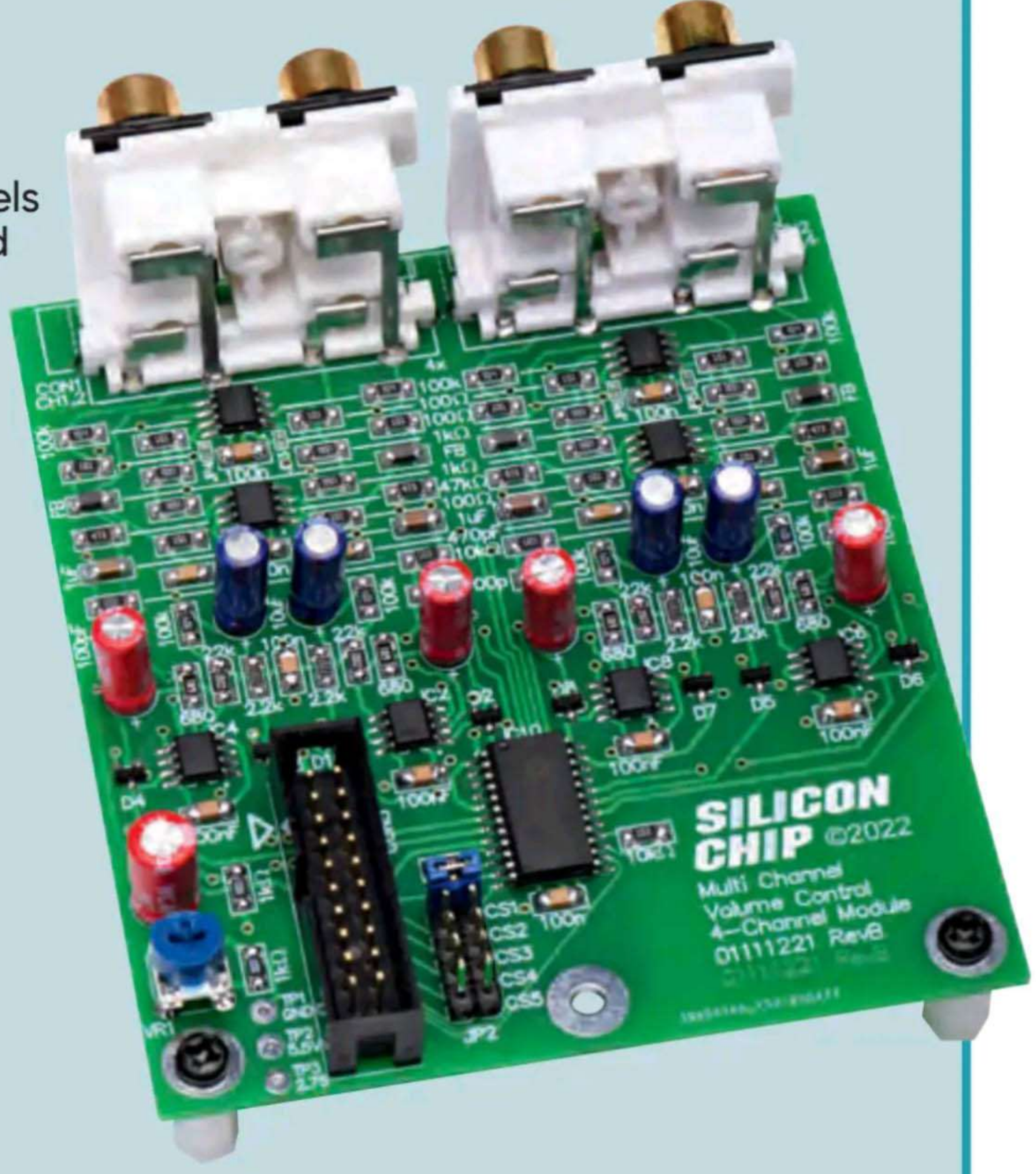
Coin Cell Emulator

Test power and current consumption with varying supply voltage

Multi-channel Digital Volume Control

Control up to 20 audio channels using a touchscreen, infrared remote or a rotary encoder





PLUS! All your favourite regular columns from *Cool Beans* and *Circuit Surgery*, to *Audio Out*, *Teach-In*, *Techno Talk* and *Net Work*.

On sale 3rd of October, 2024

Content may be subject to change

Practical Electronics

NEW subscriptions hotline!

We have recently changed the way we sell and renew subscriptions. To subscribe or renew online, paying by credit/debit card or PayPal, please visit our shop at: www.electronpublishing.com

For those wishing to pay by direct debit, cheque or over the telephone, please contact 'Select Publisher Services' by calling 01202 087631, or send a cheque (payable to: 'Practical Electronics') with your details to:

Practical Electronics Subscriptions, PO Box 6337, Bournemouth BH1 9EH, United Kingdom

Remember, we print the date of the last issue of your current subscription in a box on the address sheet that comes with your copy.

Digital subscribers, please call 01202 880299 or visit: www.electronpublishing.com

Published on approximately the first Thursday of each month by Silicon Chip Publications, U1 234 Harbord Road Brookvale NSW 2100 Australia. Printed in England by The Manson Group, Reynolds House, 8 Porters Wood, Valley Road Industrial Estate, St Albans AL3 6PZ. Distributed by Seymour, 86 Newman St, London W1T 3EX. Subscriptions UK: £33.99 (6mo); £59.99 (12mo.); £114.99 (2yrs). EUROPE: airmail service, £38.99 (6mo); £69.99 (12mo); £129.99 (2yrs). REST OF THE WORLD: airmail service, £46.99 (6mo); £84.99 (12mo); £164.99 (2yrs). Make payments to 'Practical Electronics', Practical Electronics Subscriptions, PO Box 6337, Bournemouth BH1 9EH United Kingdom. Email: pesubs@selectps.com. PRACTICAL ELECTRONICS is sold subject to the conditions that it shall not, without the written consent of the Publishers first having been given, be lent, resold, hired out or otherwise disposed of by way of Trade at more than the recommended selling price on the cover, and that it shall not be lent, resold, hired out or otherwise disposed of in a mutilated condition or in any unauthorised cover by way of Trade or affixed to or as part of any publication or advertising, literary or pictorial matter.

Did you know our online shop now sells the current issue of *PE* for £5.99 inc. p&p?

PE Practical Electronics
The UK's premier electronics, computing and maker magazine
Circuit Surgery
Digital-to-analog converter outputs
Teach-In 2024
Learn electronics with the ESP32, Part 8 – Where Am I? (GPS/GNSS)
Audio Out
More on electronic audio switching
Programming SMD micros
2m band VHF Test Signal Generator
Easy-to-build 500W Class-D Mono Amplifier!
PLUS!
Techno Talk – Sticking the landing (of a model rocket)
Cool Beans – Gray codes and more on BCD handling
Net Work – Low-cost smart watch review
Oct 2024 £5.99
www.electronpublishing.com
@practicalelec
practicalelectronics
Practical Electronics
Audio Out
Real electronic audio switching
Programming adaptor for modern PICs
Sep 2024 £5.99

You read that right! We now sell the current issue of your favourite electronics magazine for exactly the same price as in the High Street, but we deliver it straight to your door – and for UK addresses we pay the postage. No need to journey into town to queue outside the newsagent. Just go to our website, set up an account in 30 seconds, order your magazine and we'll do the rest.

www.electronpublishing.com

MICROCHIP
RNBD451PE

The Power of Choice

Beginner or Expert, Bluetooth® Solutions for All

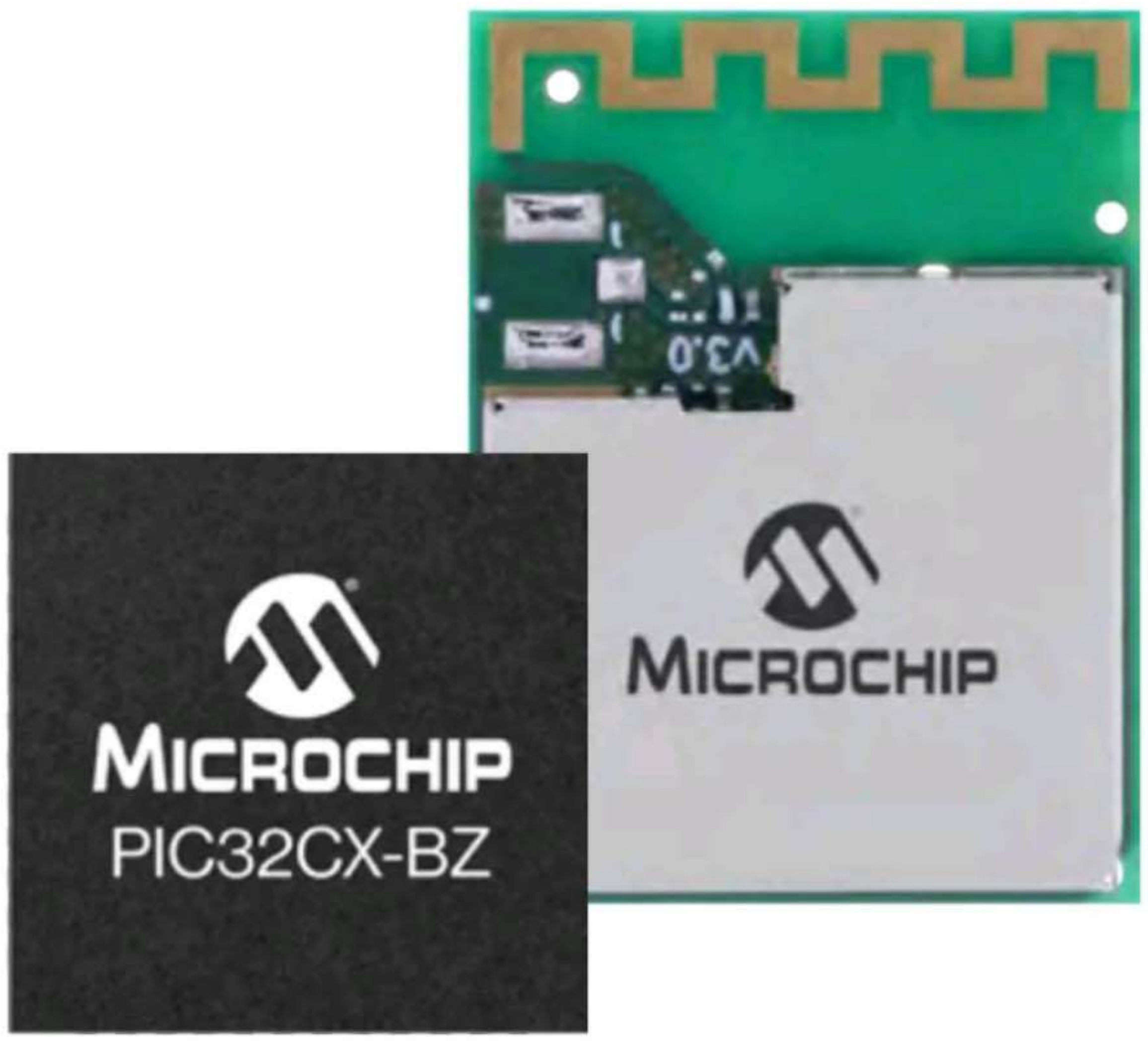
Adding Bluetooth to your product adds functionality while improving the user experience. However, finding the right provider can be difficult.

Most vendors offer an overwhelming number of products with little guidance to help you choose between them. There are so many choices that it is hard to feel confident in your selection.

Whether you are an experienced professional or new to Bluetooth development, we have a solution tailored to your needs. Our expanded Bluetooth product portfolio provides a range of options that are highly functional and user-friendly.

- **Plug-and-play modules** that require no programming or RF expertise
- **Programmable modules** that are regulatory certified multi-protocol MCUs with Bluetooth and 802.15.4
- **Robust SoCs** that are 125 °C qualified, peripheral rich and well suited for tough applications
- **Tools and support** to help you find the right part and quickly move from concept to prototype

Browse our portfolio to find the right certified Bluetooth device for you.



MICROCHIP

microchip.com/bluetooth

The Microchip name and logo and the Microchip logo are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries. All other trademarks are the property of their registered owners.
© 2024 Microchip Technology Inc. All rights reserved.
MEC2578A-UK-07-24